COURSE SCHEME AND SYLLABUS

FOR

B.E. (COE)

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

2023

SEMESTER-I

S. No.	Course Code	Course Name	CODE**	L	T	P	Cr
1.	UCB009	Chemistry	BSC	3	0	2	**4**
2.	UES103	Programming for Problem Solving	ESC	3	0	2	**4**
3.	UES013	Electrical & Electronics Engineering	ESC	3	1	2	**4.5**
4.	UEN008	Energy and Environment	OTH	2	0	0	**2**
5.	UMA010	Mathematics – I	BSC	3	1	0	**3.5**
		TOTAL					**18**

SEMESTER-II

S. No.	Course Code	Course Name	CODE**	L	T	P	Cr
1.	UPH013	Physics	BSC	3	1	2	**4.5**
2.	UES101	Engineering Drawing	ESC	2	4	0	**4**
3.	UHU003	Professional Communication	HSS	2	0	2	**3**
4.	UES102	Manufacturing Processes	ESC	2	0	2	**3**
5.	UMA004	Mathematics–II	BSC	3	1	0	**3.5**
		TOTAL					**18**

SEMESTER-III

S. No.	Course Code	Course Name	CODE**	L	T	P	Cr
1.	UCS303	Operating System	PCC	3	0	2	**4**
2.	UTA018	Object Oriented Programming	PCC	3	0	2	**4**
3.	UCS301	Data Structures	PCC	3	0	2	**4**
4.	UCS405	Discrete Mathematical Structures	PCC	3	1	0	**3.5**
5.	UES021	Engineering Materials	ESC	2	0	2	**3**
6.	UMA021	Numerical Linear Algebra	BSC	3	0	2	**4**
7.	UHU050	Evolutionary Psychology (1 Self Effort Hour)	HSS	1*	0	0	**1**
		TOTAL					**23.5**

Note: *Alternate week

SEMESTER-IV

S. No.	Course Code	Course Name	CODE**	L	T	P	Cr
1.	UCS415	Design and Analysis of Algorithms	PCC	3	0	2	**4**
2.	UCS310	Database Management Systems	PCC	3	0	2	**4**
3.	UCS503	Software Engineering	PCC	3	0	2	**4**
4.	UCS419	Artificial Intelligence	PCC	2	0	2	**3**
5.	UCS410	Probability and Statistics	BSC	3	0	2	**4**
6.	UCS418	Computer Graphics	PCC	2	0	2	**3**
7.	UTD002	Employability Development Skills	HSS	2	0	0	**2**
		TOTAL					**24**

Generic Elective

S.No.	Course No.	Course Name	L	T	P	Cr
1	UHU016	Introductory Course In French	2	0	0	2.0
2	UHU017	Introduction To Cognitive Science	2	0	0	2.0
3	UHU018	Introduction To Corporate Finance	2	0	0	2.0
4	UCS002	Introduction To Cyber Security	2	0	0	2.0
5	UPH064	Nanoscience And Nanomaterials	2	0	0	2.0
6	UEN006	Technologies For Sustainable Development	2	0	0	2.0
7	UMA069	Graph Theory And Applications	2	0	0	2.0
8	UBT510	Biology For Engineers	2	0	0	2.0

Total Credit Score for specific Nature of course/s

Nature of the course	CODE	Total Credits	Semester and Course Name
Basic Science Courses	BSC	*27*	1, Applied Chemistry (4) 1, Mathematics – I (3.5) 2, Applied Physics (4.5) 2, Mathematics–II (3.5) 3, Numerical Linear Algebra (4) 4, Probability and Statistics (3.5) 6, Optimization Techniques (4)
Engineering Science Courses	ESC	18.5	1, Programming for Problem Solving (4) 1, Electrical & Electronics Engineering (4.5) 2, Engineering Drawing (4) 2, Manufacturing Processes (3) 3, ENGINEERING MATERIALS (3)
Humanities and Social Science Courses	HSS	9	2, Professional Communication (3) 7, Humanities for engineers (3) 3, Evolutionary Psychology 4, Employability Development Skills
Professional Core Courses	PCC	58	3, Operating System (4) 3, Object Oriented Programming (4) 3, Data Structures (4) 3, Discrete Mathematical Structures(3.5) 4, Design and Analysis of Algorithms (4) 4, Database Management Systems (4) 4, Software Engineering (4) 4, Computer Architecture and Organization (3) 4, Computer Graphics (3) 5, Computer Networks (4) 5, Machine Learning (4) 5, Cognitive Computing (2) 5, Artificial Intelligence (3) 5, Theory of Computation (3.5) 6, Quantum Computing (4)

Nature of the course	CODE	Total Credits	Semester and Course Name
			7, Compiler Construction (4)
Professional Elective Courses	PEC	*12*	5, EFB-1(3) 6, EFB-II (3) 6, EFB-III (3) 7,EFB-IV (3)
Open Elective Courses	OEC	*5*	6, Innovation and Entrepreneurship (3) 6, Generic Elective (2)
Project	PRJ	29	5, Engineering Design Project-1 (3) 6, Engineering Design Project-II (3) 7, Capstone Project (8) 8, Project Semester (15)
Others	OTH	*2*	1, Energy and Environment (2)
Total		161	

UES103: Programming for Problem Solving

	L	T	P	Cr
	3	0	2	4.0

Course Objectives: This course is designed to solve and explore the problems using the art of computer programming with the help of C Language. Students will be able to apply these problem solving concepts in real life applications.

Syllabus

Introduction to Computer Fundamentals- Computer Memory Hierarchy, Types of Software Binary number system, Algorithm, Flowchart, Formulate simple algorithms for logical and arithmetic problems.

Basics of C Programming: Structure and Life cycle of a C Program, Data types, Identifiers, Variables, Keywords, Constants, input/output statements, Operators, Type conversion and type casting. Translate the algorithms to code snippets.

Decision Making and Iterative Statements-Decision making- if, if-else, Nested if-else, Multiple if, else if, switch, Ternary Operator, **Loops-** (while, do-while, for), Nesting of Loops, break, continue and goto. Implement the switch () to solve the basic functions of scientific calculator.

Functions: Function prototype, Definition and Call, Type of Functions, Scope of variables in (Block, Function, Program, File), Storage classes (Auto, Register, Static and Extern), Recursion (with the introduction of Stack), Implementation of recursion to solve the problem of Tower of Hanoi.

Arrays and Strings- One-dimensional array its operations (Traversal, Linear Search, Insertion, Deletion, Bubble Sort), Two-dimensional and its operations (Addition, Transpose and Multiplication), Passing of array into a function (row and entire array), Input and output of a string, string inbuilt functions, 2-D Character array.

Pointers: Introduction to Pointers, Pointer arithmetic, Passing arguments to a function using pointer (understanding of call by value and call by reference), Accessing arrays using pointers Dynamic memory allocation (malloc(), calloc(), realloc() and free()), Pointer and Functions.

Structures and Union: Structure declaration, Initialization of structures, Structure variables, Accessing structure elements using (.) operator, Array of structure variables, Passing structure variable to a function (individual and entire structure), Structure pointer, Comparison of Structure and Union.

File Handling: Introduction of Files (streams in C), using File (Declaring, Opening and Closing), Operations on File (Reading, Writing and appending), and Random Access of a file, command line argument.

Laboratory Work

To implement programs for various kinds of real life applications in C Language.

Course Learning Outcomes (CLOs) /Course Objectives (COs):

On completion of this course, the students will be able to:

1. Comprehend and analyze the concepts of number system, memory, compilation and debugging of the programs in C language.
2. Analyze the control & iterative statements to solve the problems with C language source codes.
3. Design and create programs for problem solving involving arrays, strings and pointers.
4. Evaluateandanalyzetheprogrammingconceptsbasedonuserdefinedatatypesandfilehandlingusing C language.

Text Books

1. CProgrammingLanguage, BrianW.KernighanDennisM. Ritchie,2nded,2012.
2. ProgramminginANSIC, BalagurusamyG., 8thed., 2019

Reference Books

1. LetUs C, KanetkarY.,16th ed.,2017
2. Programming with C, Byron S Gottfried,McGraw Hill Education, Forth edition, 2018

UCS303:Operating System

L	T	P	Cr
3	0	2	4.0

Course Objective: To understand the role, responsibilities, and algorithms involved for achieving various functionalities of an Operating System.

Syllabus

Introduction and Operating System Structures: Computer-System Organization, Computer-System Architecture, Operating-System Structure, Operating-System Operations, Computing Environments, Operating-System Services, User and Operating-System Interface, System Calls, Types of System Calls, System Programs, Operating-System Structure, System boot.

Process Management: Process Concept, Process Scheduling, Operations on Processes, Inter-process Communication, Overview of Threads, Multi-core Programming, Multithreading Models, CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling.

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Memory Management: Basic Hardware, Address Binding, Logical and Physical Address, Dynamic linking and loading, Shared Libraries, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table, Virtual Memory Management: Demand Paging, Page Replacement, Allocation of Frames, Thrashing.

Storage Management: Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, RAID Structure; File Concept, Access Methods, Directory and Disk Structure, File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods.

Protection and Security: Principles of Protection, Domain of Protection, Access Matrix, Implementation of the Access Matrix, The Security Problem, Program Threats.

Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Overview of Monitors.

Laboratory Work

Learn and practice basic Linux/Unix commands to Create and manipulate files and directories; Explore about Vi Editor environment; Build .C program related to fork (), exec (), wait (), sleep () functions at Linux/Unix platform; Write .C program for message passing and shared memory; Simulate CPU scheduling algorithms using either C or C++

Course Learning Objectives (CLO)

The students will be able to:

1. Describe the basics of an operating system, including the kernel, system calls, and computing environments.

2. Evaluate the effectivenessand trade-offs of different models of multithreading, scheduling algorithms, and methods for handling deadlocks, such as prevention, avoidance, detection, and recovery.

3. Understand components of a memory system, virtual memory and analyze different memory management techniques.

4. Evaluate the effectiveness of different Disk Management strategies, and Critique thedesign and implementation of File System

5. Explain the basic concepts of Concurrency, Protection and Security issues in an operating system.

Text Books

1. Operating System Concepts, Silberschatz A., Galvin B. P. and Gagne G., John Wiley & Sons Inc., 9th ed, 2013.
2. Operating Systems Internals and Design Principles, Stallings W., Prentice Hall 9th ed, 2018

Reference Books

1. Understanding the Linux Kernel, Bovet P. D., Cesati M., O'Reilly Media, 3rd ed, 2006.
2. Introduction to Operating System Design and Implementation: The OSP 2 Approach, Kifer M., Smolka A. S., Springer, 2007

UTA018: Object Oriented Programming

	L	T	P	Cr
	3	0	2	4.0

Course Objective:To become familiar with object oriented programming concepts and be able to apply these concepts in solving diverse range of applications.

Objects and Classes: Structure in C and C++, Class specification, Objects, Namespaces, Overview of pillars of OOPS (Data Encapsulation, Data Abstraction, Inheritance, Polymorphism), Inline functions, Passing objects as arguments, Returning object from a function, Array of objects, Static keyword with data member, member function and object, Friend function, and Friend classes, Pointer to objects, this pointer, Dynamic Initialization, Dynamic memory allocation.

Constructor and Destructor: Constructors and its types, Constructor Overloading, Constructors in array of objects, Constructors with default arguments, Dynamic Constructor, Destructor, 'const' keyword with data member, member function and object.

Inheritance: Introduction to Inheritance, Forms of Inheritance (Single, Multiple, Multilevel, Hierarchical and Hybrid) with various modes (Public, Private and Protected), Inheritance with Constructor and Destructor, Benefits and Limitations of Inheritance.

Polymorphism: Classification of Polymorphism (Compile-time and Run-time), **Compile Time-**Function Overloading, Operator Overloading (Unary operator and Binary operator with member function and friend function), Data Conversion (Basic to user-defined, user-defined to basic, one user-defined to another user-defined). **Run-time-** Pointers to derived class object, Overriding member function, Virtual functions, pure virtual functions, Abstract class.

Exception Handling, Templates and Standard Template Library: Exception handling mechanism, Usage of template, Function templates, Overloading of Function templates, Class templates, Introduction to Standard Template Library and its components. Algorithms, Containers (Array, Vector, Stack, List and Queue) and Iterators.

Laboratory Work

To implement object oriented constructs using C++programming language.

Course Learning Objectives (CLO)

The students will be able to:

1. To recall the knowledge of structure and its variables to comprehend the concept of classes, objects, constructors and destructors for implementing the object oriented paradigms.

2. To apply and analyze the inheritance on real life case studies via coding competences.
3. To design and develop code snippets for polymorphism to proclaim coding potential; and management of run-time exceptions.

4. To assess and interpret the knowledge of templates to appraise the standard template libraries.

Text Books

1. C++:The Complete Reference , Schildt H., Tata McGraw Hill, 4th ed, 2003
2. C++Primer, Lippman B.S., Lajoie J., and MooE.B., , Addison-Wesley Professional, 5th ed, 2013

Reference Books

1. Object-Oriented Programming in C++, Lafore R., Pearson Education, 4th ed, 2002
2. Object Oriented Programming with C++, E Balagurusamy, 8th ed,2017
3. The C++programming language, Stroustrup B., Pearson Education India, 4th ed, 2013

UCS301:**Data Structures**

	L	T	P	Cr
	3	0	2	4.0

Course Objective:To become familiar with different types of data structures and their applications.

Syllabus

Analysing algorithms: Basics of algorithm and its analysis, Complexity classes, order arithmetic, Time and space trade-off in algorithms.

Linear Data Structures: Arrays, Strings and string processing, Linked lists (Singly, Doubly, Circular), Abstract data types, their implementation and applications: Stacks (using Arrays and Linked-list), Queues (using Arrays and Linked-list), Hash tables: Hash functions, collision resolution techniques, Strategies for choosing the appropriate data structure.

Searching and Sorting: Linear Search, Binary Search. Introduction to internal and external sort, Bubble Sort, Selection Sort, Insertion Sort, Shell Sort, Quick Sort, Merge Sort, Counting Sort, Radix Sort.

Trees and their applications: Introduction to binary tree, tree traversal algorithms, Binary search tree, AVL Tree, B Tree etc. and common operations on these trees. Heap, Heap Sort, Priority Queue using Heap.

Graphs and their applications: Graph Terminology and its representation, Depth and breadth first traversals, Shortest-path algorithms (Dijkstra and Floyd), Data Structures for Disjoint Sets, Minimum spanning tree (Prim and Kruskal).

Laboratory Work

Implementation of various data structures such as Arrays, Stacks, Queues, Lists, Binary tree traversals, BST, AVL trees, Graphs traversals, Sorting and Searching techniques.

Course Learning Objectives (CLO)

The students will be able to:

1. Understand the fundamental data structures, their implementation and some of their standard applications.

2. Select and implement appropriate searching and sorting techniques for solving a problem based on their characteristics.

3. Apply tree and graph data structures for specific applications.

4. Design and analyse algorithms using appropriate data structures for real-world problems.

Text Books

1. Introduction to Algorithms,Cormen H. T., Leiserson E. C., Rivest L. R., and Stein C, MIT Press,3rd ed., 2009
2. Data Structures, Algorithms and Applications in C++,Sahni S., Universities Press 2nd ed. 2005

Reference Books

1. Data Structures and Algorithms Made Easy,Karumanchi N., Career Monk Publications, 5th ed., 2017
2. Data structures and algorithms in C++, Adam Drozdek, 4th edition.

UCS405:Discrete Mathematical Structures

	L	T	P	Cr
	3	1	0	3.5

Course Objective:The course objective is to provide students with an overview of Discrete Mathematical Structures. Students will learn about topics such as logic and proofs, sets and functions, graph theory, boolean algebra, number theory and other important discrete math concepts.

Syllabus

Sets, Relations, and Functions: Sets: Operations on set, Inclusion-exclusion principle, Representation of Discrete Structures, Fuzzy set, Multi-set, bijective function, Inverse and Composition of functions, Floor and Ceiling functions, Growth of functions: Big-O notation, Big-Omega and Big-Theta Notations, Determining complexity of a program, Hash functions.

Relations: Different types of relation and their representation, Equivalence and partial-ordered relations, Partition and Covering of a set, N-ary relations and database, Closure of relations, Warshall's algorithm, Lexicographic ordering, Hasse diagram, Lattices, Boolean algebra.

Graphs Theory: Representation, Type of Graphs, Paths and Circuits: Euler Graphs, Hamiltonian Paths & Circuits; Cut-sets, Connectivity and Separability, Planar Graphs, Isomorphism, Graph Coloring, Covering and Partitioning, Application of Graph theory in real-life applications.

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

Recurrence Relation: Solving linear recurrence relations, divide and conquer algorithms and recurrence relations.

Algebraic Structures: Group, Semi group, Monoids, Ring, Field, Homomorphism.

Number Theory: Divisibility and Modular Arithmetic, Solving Congruences, Applications of Congruences, Cryptographic applications

Laboratory Work

NA

Course Learning Objectives (CLO)

The students will be able to:

1. Perform operations on various discrete structures such as set, function, and relation.
2. Apply basic concepts of asymptotic notation in the analysis of the algorithm.
3. Illustrate the basic properties and algorithms of graphs and apply them in modelling and solving real-world problems.
4. Comprehend formal logical arguments and translate statements from a natural language into their symbolic structures in logic.
5. Identify and prove various properties of rings, fields, and groups.
6. Illustrate and apply the division algorithm, mod function, and Congruence.

Text Books

1. Discrete Mathematics and its Applications,Rosen H. K., McGraw Hill, 7thed., 2011
2. Discrete Mathematical Structures with Applications to Computer Science, Tremblay P. J. and Manohar, R., Tata McGraw Hill, 2008.

Reference Books

1. Contemporary Abstract Algebra, Gallian A. J., Cengage Learning, 9th ed., 2017
2. Discrete Mathematics, Lipschutz S., Lipson M., McGraw-Hill, 3rd ed.,2007

UCS415:DESIGN AND ANALYSIS OF ALGORITHMS				
	L	T	P	Cr
	3	0	2	4.0

Course Objective To provide students with the knowledge and skills necessary to design and analyse algorithms for solving computational problems.

Syllabus:

Introduction and Complexity Analysis: Analysing algorithms, Complexity classes, Time and space trade-offs in algorithms, Recurrence relations, Analysis of iterative and recursive algorithms, Amortized Analysis.

Algorithm Design Techniques and Analysis

Divide and Conquer: Fundamentals of divide and conquer strategy, Applications such as The maximum subarray problem, Strassen's algorithm for matrix multiplication, merge sort, quick sort etc.

Greedy Algorithms: Elements of greedy strategy, Applications such as activity selection, Huffman Coding, job sequencing, fractional knapsack problem, etc.

Dynamic Programming: Elements of dynamic programming, Memorization and tabulation approaches, Applications such as matrix multiplication, 0/1 knapsack, Longest common subsequence, Optimal binary search tree, etc.

Backtracking: Introduction, Applications such as N queen problem, sum of subsets, graph coloring, etc.

Branch and Bound Algorithm: General method, Applications such as0/1 knapsack problem, Traveling salesperson problem etc.

Graphs & Algorithms: Introduction to graphs, Paths and Circuits, Euler Graphs, Hamiltonian graphs,Cut-sets, Connectivity and Separability, Covering and Partitioning, Strongly connected component, Topological sort, Max flow: Ford Fulkerson algorithm, max flow- min cut.

String Matching Algorithms: Suffix arrays, Rabin-Karp, Knuth-MorrisPratt (KMP), Boyer Moore algorithm.

Problem Classes: P, NP, NP-Hard and NP-complete, deterministic and non-deterministic polynomial time algorithm approximation, Randomized algorithms.

Laboratory Work (if applicable): Implementation of various algorithmic techniques for solving common computational/engineering problems.

Course Learning Objectives (CLO)

The students will be able to:

1. Analyse the complexity of algorithms and implement it in a specific scenario.
2. Apply common algorithmic techniques such as greedy, dynamic programming etc. to standard computational problems
3. Design solutions by using appropriate data structures or applying algorithms such as string matching, randomized, approximation and graph.
4. Develop efficient algorithms for various computational challenging problems solving in computing.

Text Books

1. Cormen H. T., Leiserson E. C., Rivest L. R., and Stein C., Introduction to Algorithms, MIT Press (2009) 3rd ed.
2. Horwitz E., Sahni S., Rajasekaran S., Fundamentals of Computers Algorithms, Universities Press (2008) 2nd ed.

Reference Books

1. Levitin A., Introduction to the design and analysis of algorithms, Pearson Education (2008) 2nd ed.
2. Aho A.V., Hopcraft J. E., Dulman J. D., The Design and Analysis of Computer Algorithms, Addsion Wesley (1974) 1st ed.
3. Sedgewick R. and Wayne K., Algorithms, Addison-Wesley Professional (2011), 4th ed.

UCS419: Artificial Intelligence				
	L	T	P	Cr
	2	0	2	3.0

Course Objectives: This course introduces students to the fundamental concepts, techniques, and applications of Artificial Intelligence (AI). Students will gain theoretical knowledge and practical skills in areas such as problem-solving using search techniques, machine learning and designing intelligent agents for solving particular engineering problems.

Syllabus:

Introduction to Artificial Intelligence: Foundations, scope, types of AI, problems, and approaches of AI

Intelligent agents: Structure of agents, Types of agent programs: reflux, model-based, goal-driven, utility-driven, and learning agents

Problem spaces: State Space Representation, Representation of problems as state space, problem characteristics, sample applications

Uninformed Search Algorithms: Brute Force search, Depth-First Search, Breadth-First search, Depth-Limited Search, Uniform Cost Search, Bidirectional Search

Informed search algorithms: Heuristic Functions, Best-First search, Beam Search, Hill Climbing, A* algorithm, AO graph, stochastic search algorithms: Simulated Annealing and Genetic Algorithm

Game playing: Minimax algorithm, alpha-beta pruning, iterative deepening

Introduction to Machine Learning: Well-Posed learning problems, Basic concepts, Designing a learning system, Types of machine learning: Supervised learning, Unsupervised learning, Semi-supervised Learning and Reinforcement learning, Types of data: structured and unstructured data.

Supervised Learning: Introduction to supervised learning tasks, Tree induction algorithms: split algorithm based on Information Gain (ID3), split algorithm based on Gain Ratio (C4.5), split algorithm based on Gini Index (CART), Instance based algorithms: K-Nearest Neighbours (K-NN), Probabilistic algorithms: Naïve Bayes algorithm, Evaluation metrics.

Unsupervised Learning: Introduction to supervised learning tasks, Partitioning-based methods

Laboratory Work (if applicable): Basics of Python programming language: Data Types, Data Structures, Flow Control, Functions, Basic Data Science packages: NumPy, Pandas, SciPy

Implementing Search algorithms in C/C++/Java/Python: Depth first, Breadth first, Hill

climbing, best first, A* algorithm, Implementation of games: 8-puzzle, Tic-Tac-Toe, tower of Hanoi and water jug problem using heuristic search

Implementing Machine Learning algorithms: Tree-based methods, K-NN, Naïve-Bayes algorithms, K-Means (from scratch and using sklearn library)

Course Learning Objectives (CLO)

The students will be able to:

1. Analyze methods and theories in the field of Artificial Intelligence and categorize various problem domains.

2. Design intelligent agents for concrete computational problems.

3. Analyze and apply different problem-solving strategies and search algorithms.

4. Implement and evaluate machine learning algorithms for various real-world tasks.

Text Books

1. Russel S., Norvig P., Artificial Intelligence: A Modern Approach, Prentice Hall (2014) 3rd ed.

2. Murphy, Kevin P. Machine learning: a probabilistic perspective. MIT press, (2012) 3rd ed.

Reference Books

1. Rich E., Knight K. and Nair B. S., Artificial Intelligence, Tata McGraw Hills (2009) 3rd ed.

2. Luger F. G., Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education Asia (2009) 6th ed.

UCS310:Database Management Systems

	L	T	P	Cr
	3	0	2	4.0

Course Objective:Emphasis is on the need of database systems. Main focus is on E-R diagrams, relational database, concepts of normalization and de-normalization and SQL commands.

Syllabus

Introduction: Data, data processing requirement, desirable characteristics of an ideal data processing system, traditional file-based system, its drawback, concept of data dependency, Definition of database, types of database, database management system, 3-schema architecture, database terminology, benefits of DBMS.

Relational Database: Relational data model: Introduction to relational database theory: definition of relation, keys, relational model integrity rules, introduction to Relational Algebra.

Database Analysis: Conceptual data modeling using E-R data model -entities, attributes, relationships, generalization, specialization, specifying constraints, Conversion of ER Models to Tables, Practical problems based on E-R data model.

Database Design: Functional Dependency, Canonical Covers, Candidate Key Identification, Normalization- 1NF, 2NF, 3NF, BCNF, 4NF and 5NF. Concept of De-normalization and practical problems based on these forms.

Transaction Management and Concurrency control: Concept of Transaction, States of Transaction and ACID properties, Need of Concurrency control, concept of Lock, Two phase locking protocol.

Recovery Management: Need of Recovery Management, Concept of Stable Storage, Log Based Recovery Mechanism, Checkpoint.

Database Implementation: Introduction to SQL, DDL aspect of SQL, DML aspect of SQL – update, insert, delete & various form of SELECT- simple, using special operators, aggregate functions, group by clause, sub query, joins, co-related sub query, union clause,

View, exist operator. PL/SQL - cursor, stored function, stored procedure, triggers, error handling, and package.

Laboratory Work

Students will perform SQL commands to demonstrate the usage of DDL and DML, joining of tables, grouping of data and will implement PL/SQL constructs. They will also implement one project.

Project: It will contain database designing & implementation, should be given to group of 2-4 students. While doing projects emphasis should be more on back-end programming like use of SQL, concept of stored procedure, function, triggers, cursors, package etc. Project should have continuous evaluation and should be spread over different components.

Course Learning Objectives (CLO)

The students will be able to:

1. Analyze the Information Systems as socio-technical systems, its need and advantages as compared to traditional file-based systems.

2. Analyze and design database using E-R data model by identifying entities, attributes and relationships.

3. Apply and create Relational Database Design process with Normalization and Denormalization of data.

4. Comprehend the concepts of transaction management, concurrence control and recovery management.

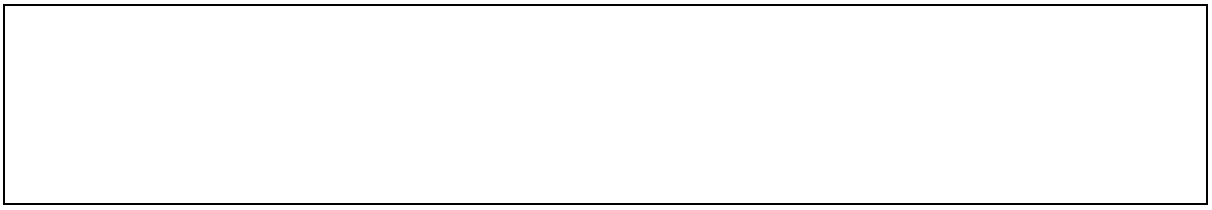
5. Demonstrate use of SQL and PL/SQL to implementation database applications.

Text Books

1. Database System Concepts, Silverschatz A., Korth F. H. and Sudarshan S., Tata McGraw Hill, 6th ed, 2010
2. Fundamentals of Database Systems,Elmasri R. and Navathe B. S., Pearson, 7th ed, 2016

Reference Books

1. SQL, PL/SQL the Programming Language of Oracle, Bayross I., BPB Publications, 4th ed, 2009
2. Modern Database Management,Hoffer J., Venkataraman, R. and Topi, H., Pearson, 12th ed2016
3. Simplified Approach to DBMS, Parteek Bhatia and Gurvinder Singh,
4. Database management systems. Vol. 3. Raghu Ramakrishnan and Johannes Gehrke
5. FOR SQL/RA, New York: McGraw-Hill,



UCS503:SOFTWAREENGINEERING				
	L	T	P	Cr
	3	0	2	4.0

Course Objective To plan and manage large scale software and learn emerging trends insoftwareengineering.

Syllabus:

SoftwareEngineeringandProcesses: Introduction to Software Engineering, Software Evolution, Software Characteristics, Software Crisis: Problems and Causes, Software Myths, Software process models-Waterfall, Iterative, Incremental and Evolutionary process models, Unified Process Model

RequirementsEngineering: Problem Analysis, Structured Analysis Vs Object Oriented Analysis Requirement Elicitation and Validation, Requirement Analysis Approaches-FlowmodelingthroughDataFlowDiagramandDataDictionarydocumentingSoftwareRequirement Specification (SRS)

SoftwareDesignandconstruction: Systemdesignprincipleslikelevelsofabstraction,separation of concerns, information hiding, coupling and cohesion, Structured design (top-downorfunctionaldecomposition),object-orienteddesign,eventdrivendesign,component-leveldesign, architecture design like Model ViewController, Client – Server architecture, Software Design using UML: Use Case Diagram, Swimlane Diagram, Sequence and Collaboration Diagram, Class and Object Diagram, State Machine Diagram, Component and Deployment Diagram.

SoftwareVerificationandValidation: LevelsofTesting,FunctionalTesting,StructuralTesting,TestPlan,TestCaseSpecification,SoftwareTestingStrategies,Verification&Validation, Unit and Integration Testing, Alpha & Beta Testing, White box and black box testingtechniques,System TestingandOverview ofDebugging.

Agile Software Development: Agile Manifesto, Twelve Practices of eXtreme Programming(XP), XPvalues,XPpractices,project velocityworkingof Scrum,productbacklog,sprintbacklog, FeatureDrivenDevelopment(FDD),TestDrivenDevelopment, Agile Requirement and Design: User Stories, Story Boards, UI Sketching and StoryCards.

SoftwareProjectManagement: OverviewofProjectManagement:Estimation of Scope, Time and Cost of Project, Scope estimation – Function Point Count vs LOC,Time Estimation –

Critical Path Method andCost Estimation – COCOMO model

Laboratory Work (if applicable): Implementation of Software Engineering concepts and exposure to CASE tools like RationalSoftwareSuit through projects.

Course Learning Objectives (CLO)

The students will be able to:

1. Analyzesoftwaredevelopmentprocessmodelsforsoftwaredevelopmentlifecycle.
2. Elicit,describe,andevaluateasystem'srequirementsandanalyzethemusingvariousUML models.
3. Demonstratetheuseofdesignprinciplesindesigningdata,architecture,userandcomponent level design.
4. Testthesystembyplanningappropriatetest casesandapplyingrelevantteststrategies.
5. ComprehendtheuseofagiledevelopmentmethodologiesincludingUIsketching,userstorie s,storycardsand backlogmanagement.

Text Books

1. Pressman R., Software Engineering, A Practitioner‘s Approach, McGraw Hill International,7th ed.(2010).
2. SommervilleI.,SoftwareEngineering,Addison-WesleyPublishingCompany,9th ed.(2011).

Reference Books

1. JaloteP.,AnintegratedApproach toSoftwareEngineering,Narosa, 3rd ed.(2005).
2. BoochG.,RambaughJ.,JacobsonI.,TheUnifiedModelingLanguageUserGuide,2nd ed.(20 05).

UCS418: COMPUTER GRAPHICS				
	L	T	P	Cr
	2	0	2	3.0

Course Objective This course provides an introduction to the principles of computer graphics. It covers detailed study of computer graphics fundamentals, 2-D & 3-D geometric transformations, curve design, and visible surface detection.

Syllabus:

Fundamentals of Computer Graphics: Applications of computer Graphics in various fields, Evolution of computer Graphics, Graphical Input-Output Devices, Random scan displays, Raster scan displays.

Graphics Primitives: Algorithms for drawing various output primitives - Line, circle, ellipse, Boundary Fill & Flood Fill algorithm.

2-D & 3-D Geometrical Transformations: Translation, Rotation, Scaling, Shear, Reflection, Homogenous coordinate system, Composite transformations.

Viewing &Clipping in 2-D: Window to View port transformation, Cohen Sutherland, Liang Barsky, Sutherland Hodgeman,clipping algorithm.

Three Dimensional Viewing & Clipping:3-D Viewing, Projections, Parallel and Perspective projections.

Curves& Surfaces: Curved Lines & surfaces, Interpolation & Approximation splines, Parametric & Geometric Continuity conditions, Bezier Curves & surfaces, B-spline curves & surfaces.

Visible Surface Detection Methods: Classification of visible surface detection algorithms, Depth buffer method, Depth-Sorting method.

Laboratory Work (if applicable): Laboratory work should be done in OpenGL (version 3+). Covers all the basic drawing (output primitives in OpenGL), shape drawing

algorithms, filling, 2D & 3D transformations, clipping.

Course Learning Objectives (CLO)

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

1. Comprehend the concepts related to basics of computer graphics and its applications in various fields.
2. Apply algorithms to scan convert various output primitives and alters the coordinate descriptions of objects using 2-D & 3-D geometric transformations.
3. Understand and apply various concepts of viewing & clipping.
4. Comprehend the concepts related to curve design and identify visible surfaces in three dimensional scene using visible surface detection methods.
5. Apply OpenGL to create various primitives of computer graphics.

Text Books

1. Donald D Hearn, M. Pauline Baker, "Computer Graphics, C version", 2nd Edition, Pearson Education (1997).
2. James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes, "Computer Graphics: Principles & Practice in C", Second Edition, Addison Wesley Longman (1995).

Reference Books

1. Donald Hearn and M Pauline Baker, "Computer Graphics with OpenGL", Pearson education, 2007.
2. Zhigang Xiang, Roy A Plastock, "Computer Graphics", Schaums Outline, TMH (2007).
3. Dave Shreiner, Mason Woo, Jackie Neider, Tom Davis, "OpenGL Programming Guide: The Official Guide to Learning OpenGL" (2013).

UHU006: INTRODUCTORY COURSE IN FRENCH

L	T	P	Cr
2	0	0	2.0

Course Objectives:

The objectives of the course are to introduce to the students:

1. The basics of French language to the students. It assumes that the students have minimal or no prior knowledge of the language.
2. To help them acquire skills in writing and speaking in French, comprehending written and spoken French.
3. The students are trained in order to introduce themselves and others, to carry out short conversation, to ask for simple information, to understand and write short and simple messages, to interact in a basic way.
4. The main focus of the students will be on real life language use, integration of French and francophone culture, & basic phrases aimed at the satisfaction of needs of concrete type.
5. During class time the students are expected to engage in group & pair work.

Communicative skills: Greetings and Its Usage, Asking for and giving personal information, How to ask and answer questions, How to talk over the phone, Exchange simple information on preference, feelings etc. Invite, accept, or refuse invitation, Fix an appointment, Describe the weather, Ask for/give explanations, Describe a person, an object, an event, a place.

Grammar : Pronouns: Pronom sujets (Je/ Tu/Il/Elle/Nous/Vous/Ils/Elles), Nouns: Genders, Articles: Definite article and Indefinite articles, Verbs: Regular verbs (-er, -ir ending) Irregular verbs (-re ending), Auxiliary verbs (avoir, être, aller). Adjective: Description, Adjective possessive, Simple Negation, Tense: Present, Future, Questions, Singular & plural.

Vocabulary: Countries and Nationalities, Professions, Numbers (ordinal, cardinal), Colours, Food and drinks, Days of the week, Months, Family, Places.

Phonetics: The course develops the ability, to pronounce words, say sentences, questions and give orders using the right accent and intonation. To express surprise, doubt, fear, and all positive or negative feelings using the right intonation. To distinguish voiced and unvoiced consonants. To distinguish between vowel sounds.

Course Learning Outcomes (CLOs):

Upon the completion of the course:

1. The students begin to communicate in simple everyday situations acquiring basic grammatical structure and vocabulary.
2. The course develops oral and reading comprehension skills as well as speaking and writing.

3. Students can demonstrate understanding of simple information in a variety of authentic materials such as posters, advertisement, signs etc.
4. Discuss different professions, courses and areas of specialisation.
5. Write simple messages, letters, composition and dialogues. Complete simple forms and documents.
6. Express feelings, preferences, wishes and opinions and display basic awareness of francophone studies.
7. Units on pronunciation and spelling expose students to the different sounds in the French language and how they are transcribed.

Recommended Books :

1. *Alter ego-1 : Méthode de français by Annie Berthet, Catherine Hugot, Véronique M. Kizirion, Beatrix Sampsonis, Monique Waendendries, Editions Hachette français langue étrangère.*
2. *Connexions-1 : Méthode de français by Régine Mérieux, Yves Loiseau, Editions Didier*
3. *Version Originale-1: Méthode de français by Monique Denyer, Agustin Garmendia.*
4. *Marie-Laure Lions-Olivieri, Editions Maison des Langues, Paris 2009*
5. *Latitudes-1 : Méthode de français by Régine Mérieux, Yves Loiseau, Editions Didier*
6. *Campus-1 : Méthode de français by Jacky Girardet, Jacques Pécheur, Editions CLE International.*
7. *Echo-1 : Méthode de français by J. Girardet, J. Pécheur, Editions CLE International.*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UCS001: INTRODUCTION TO CYBER SECURITY

L	T	P	Cr
2	0	0	2.0

Course Objectives: In this course, the student will learn about the essential building blocks and basic concepts around cyber security such as Confidentiality, Integrity, Availability, Authentication, Authorization, Vulnerability, Threat and Risk and so on.

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

Programs and Programming: Unintentional (Non-malicious) Programming Oversights, Malicious Code—Malware, Countermeasures

Web Security: User Side, Browser Attacks, Web Attacks Targeting Users, Obtaining User or Website Data, Email Attacks

Operating Systems Security: Security in Operating Systems, Security in the Design of Operating Systems, Rootkit

Network Security: Network Concepts, Threats to Network Communications, Wireless Network Security, Denial of Service, Distributed Denial-of-Service Strategic Defenses: Security Countermeasures, Cryptography in Network Security, Firewalls, Intrusion Detection and Prevention Systems, Network Management

Cloud Computing and Security: Cloud Computing Concepts, Moving to the Cloud, Cloud Security Tools and Techniques, Cloud Identity Management, Securing IaaS

Privacy: Privacy Concepts, Privacy Principles and Policies, Authentication and Privacy, Data Mining, Privacy on the Web, Email Security, Privacy Impacts of Emerging Technologies, Where the Field Is Headed

Management and Incidents: Security Planning, Business Continuity Planning, Handling Incidents, Risk Analysis, Dealing with Disaster

Legal Issues and Ethics: Protecting Programs and Data, Information and the Law, Rights of Employees and Employers, Redress for Software Failures, Computer Crime, Ethical Issues in Computer Security, Incident Analysis with Ethics

Emerging Topics: The Internet of Things, Economics, Computerized Elections, Cyber Warfare.

Course Learning Outcomes (CLOs):

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

1. Understand the broad set of technical, social & political aspects of Cyber Security and security management methods to maintain security protection
2. Appreciate the vulnerabilities and threats posed by criminals, terrorist and nation states to national infrastructure
3. Understand the nature of secure software development and operating systems
4. Recognize the role security management plays in cyber security defense and legal and social issues at play in developing solutions.

Recommended Books:

1. *Pfleeger, C.P., Security in Computing, Prentice Hall, 5th edition (2010)*
2. *Schneier, B., Applied Cryptography, Second Edition, John Wiley & Sons (1996)*
3. *Rhodes-Ousley, M., Information Security: The Complete Reference, Second Edition, Information Security Management: Concepts and Practice. New York, McGraw-Hill, (2013).*
4. *Whitman, M.E. and Herbert J. M., Roadmap to Information Security for IT and Infosec Managers, Course Technology, Boston, MA (2011).*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UTD001: EMPLOYABILITY DEVELOPMENT SKILLS

L	T	P	Cr
2	0	0	2.0

Course Objectives:

This course aims to sensitize students with the gamut of skills which facilitate them to enhance their employability quotient and do well in the professional space. These skills are imperative for students to establish a stronger connect with the environment in which they operate. An understanding of these skills will enable students to manage the placement challenges more effectively.

Emotional Intelligence: Understanding Emotional Intelligence (EI); Daniel Goleman's EI Model: Self Awareness, Self-Regulation, Internal Motivation, Empathy, Social Skills; Application of EI during Group Discussions & Personal Interview; Application of EI in personal life, student life and at the workplace

Team Dynamics & Leadership: Understanding the challenges of working within a team format in today's complex organizational environments; Stages of team formation; Appreciating forces that influence the direction of a team's behaviour and performance; Cross-functional teams; Conflict in Teams- leveraging differences to create opportunity Leadership in the team setting & energizing team efforts; Situational leadership; Application of team dynamics & collaboration in Group Discussions; Application of team dynamics at the workplace

Complex Problem Solving: Identifying complex problems and reviewing related information to develop and evaluate options and implement solutions; Understanding a working model for complex problem solving - framing the problem, diagnosing the problem, identifying solutions & executing the solutions; Appreciation of complex problem solving at the workplace through case studies

Lateral Thinking: Understanding lateral thinking & appreciating the difference between vertical & lateral thinking, and between convergent & divergent thinking; Understanding brain storming & mind-maps; Solving of problems by an indirect and creative approach, typically through viewing the problem in a new and unusual light; Application of lateral thinking during Group Discussions & Personal Interviews; Application of lateral thinking at the workplace

Persuasion: Role of persuasion in communication; Application of ethos-pathos-logos; Using persuasive strategies to connect with individuals & teams to create competitive advantage

Quantitative Reasoning: Thinking critically and applying basic mathematics skills to interpret data, draw conclusions, and solve problems; developing proficiency in numerical reasoning; Application of quantitative reasoning in aptitude tests

Verbal Reasoning: Understanding and reasoning using concepts framed in words; Critical verbal reasoning; Reading Comprehension; Application of verbal reasoning in aptitude tests

Group Discussion (GD): Illustrating the do's and don'ts in Group Discussions; Specific thrust on types of GD topics; GD evaluation parameters; Understanding the challenge in a

case discussion; SPACER model

Personal Interview (PI): Interview do's and don'ts; PI evaluation parameters; The art of introduction; Managing bouncer questions; Leading the panel in a PI

Course Learning Outcomes (CLOs):

The students will be able to

1. appreciate the various skills required for professional & personal success.
2. bridge the gap between current and expected performance benchmarks.
3. competently manage the challenges related to campus placements and perform to their utmost potential.

Recommended Books:

1. *Harvard Business Essentials; Creating Teams with an Edge; Harvard Business School Press (2004)*
2. *Edward de B., Six Thinking Hats; Penguin Life (2016)*
3. *Daniel, G., Working with Emotional Intelligence; Bantam Books (2000)*
4. *Aggarwal, R.S., Quantitative Aptitude for Competitive Examinations; S Chand (2017)*
5. *Agarwal, A., An expert guide to problem solving: with practical examples; CreateSpace Independent Publishing Platform (2016)*
6. *William, D., The Logical Thinking process; American Society for Quality (2007)*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UEN004: TECHNOLOGIES FOR SUSTAINABLE DEVELOPMENT

L	T	P	Cr
2	0	0	2.0

Course Objectives: To provide acquaintance with modern cleaner production processes and emerging energy technologies; and to facilitate understanding the need and application of green and renewable technologies for sustainable development of the Industry/society

Concepts of Sustainability and Industrial Processes: Industrialization and sustainable development; Cleaner production (CP) in achieving sustainability; Source reduction techniques - Raw material substitution; Process modification and equipment optimization; Product design or modification; Reuse and recycling strategies; Resources and by-product recovery from wastes; Treatment and disposal; CDM and Pollution prevention programs; Good housekeeping; CP audits,

Green Design: Green buildings - benefits and challenges; public policies and market-driven initiatives; Effective green specifications; Energy efficient design; Passive solar design; Green power; Green materials and Leadership in Energy and Environmental Design (LEED)

Renewable and Emerging Energy Technologies: Introduction to renewable energy technologies- Solar; wind; tidal; biomass; hydropower; geothermal energy technologies; Emerging concepts; Biomolecules and energy; Fuel cells; Fourth generation energy systems,

Course Learning Outcomes (CLOs):

Upon completion of the course, the students will be able to:

1. comprehend basic concepts in source reduction, waste treatment and management
2. Identify and plan cleaner production flow charts/processes for specific industrial sectors
3. examine and evaluate present and future advancements in emerging and renewable energy technologies

Recommended Books

1. *Kirkwood, R,C, and Longley, A,J, (Eds,), Clean Technology and the Environment, Chapman & Hall, London (1995),*
2. *World Bank Group; Pollution Prevention and Abatement Handbook – Towards Cleaner Production, World Bank and UNEP; Washington DC (1998),*
3. *Modak, P,, Visvanathan, C, and Parasnis, M,, Cleaner Production Audit, Course Material on Cleaner Production and Waste Minimization; United Nations Industrial Development Organization (UNIDP) (1995),*
4. *Rao, S, and Parulekar, B,B,, Energy Technology: Non-conventional; Renewable and Conventional; Khanna Pub,(2005) 3rd Ed,*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UHU008: INTRODUCTION TO CORPORATE FINANCE

L	T	P	Cr
2	0	0	2.0

Course Objective: This course aims to provide the students with the fundamental concepts, principles and approaches of corporate finance, enable the students to apply relevant principles and approaches in solving problems of corporate finance and help the students improve their overall capacities.

Introduction to corporate finance: Finance and corporate finance. Forms of business organizations, basic types of financial management decisions, the goal of financial management, the agency problem; the role of the financial manager; basic types of financial management decisions.

Financial statements analysis: Balance sheet, income statement, cash flow, fund flow financial statement analysis Computing and interpreting financial ratios; conducting trend analysis and Du Pont analysis.

The time value of money: Time value of money, future value and compounding, present value and discounting, uneven cash flow and annuity, discounted cash flow valuation.

Risk and return: Introduction to systematic and unsystematic risks, computation of risk and return, security market line, capital asset pricing model.

Long-term financial planning & Financial Decisions: Various sources of long term financing, the elements and role of financial planning, financial planning model, percentage of sales approach, external financing needed. Cost of capital, financial leverage, operating leverage. Capital structure, theories of capital structure net income, net operating income & M&M proposition I and II.

Short-term financial planning and management: Working capital, operating cycle, cash cycle, cash budget, short-term financial policy, cash management, inventory management, credit management.

Capital budgeting : Concepts and procedures of capital budgeting, investment criteria (net present value, payback, discounted payback, average accounting return, internal rate of return, profitability index), incremental cash flows, scenario analysis, sensitivity analysis, break-even analysis,

Dividend policy: Dividend, dividend policy, Various models of dividend policy (Residual approach, Walter model, Gordon Model, M&M, Determinants of dividend policy.

Security valuation: Bond features, bond valuation, bond yields, bond risks, stock features, common stock valuation, and dividend discount & dividend growth models. Common stock yields, preferred stock valuation.

Recommended Books:

1. *Brealey, R. A., Myers. S.C., Allen, F., Principles of Corporate Finance (9th edition), The McGraw-Hill, London, (2006).*
2. *Ehrhardt, M.C., Brigham, E.F., Financial Management: Theory and Practice (10th edition) South Western-Cengage, New York (2011)*
3. *Van Horne, J.C., Wachowicz, J.M., Kuhlemeyer, G.A., 2005, Fundamentals of Financial Management, Pearson, Vancouver (2010)*
4. *Pandey, I. M., Financial management, Vikas Publishing House Pvt. Ltd., Noida (2011)*
5. *Elton, E.J. and Gruber, M.J., Modern Portfolio Theory and Investment Analysis, (7th Edition), John Wiley and Sons, New York (2007)*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UHU017: INTRODUCTION TO COGNITIVE SCIENCE

L	T	P	Cr
2	0	0	2.0

Course Objectives: This course provides an introduction to the study of intelligence, mind and brain from an interdisciplinary perspective, It encompasses the contemporary views of how the mind works, the nature of reason, and how thought processes are reflected in the language we use, Central to the course is the modern computational theory of mind and it specifies the underlying mechanisms through which the brain processes language, thinks thoughts, and develops consciousness,

Overview of Cognitive Science: Newell's big question, Constituent disciplines, Interdisciplinary approach, Unity and diversity of cognitive science,

Philosophy: Philosophy of Mind, Cartesian dualism Nativism vs, empiricism, Mind-body problem, Functionalism, Turing Test, Modularity of mind, Consciousness, Phineas Gage, Physicalism.

Psychology: Behaviorism vs, cognitive psychology, The cognitive revolution in psychology, Hardware/software distinction , Perception and psychophysics, Visual cognition, Temporal dynamics of visual perception, Pattern recognition, David Marr's computational theory of vision, Learning and memory, Theories of learning, Multiple memory systems, Working Memory and Executive Control, Memory span, Dissociations of short- and long-term memory, Baddeley's working memory model.

Linguistics: Components of a grammar, Chomsky, Phrases and constituents, Productivity, Generative grammars, Compositional syntax, Productivity by recursion, Surface- and deep structures, Referential theory of meaning, Compositional semantics, Semantics, Language acquisition, Language and thought.

Neuroscience: Brain anatomy, Hierarchical functional organization, Decorticate animals, Neuroimaging, Neurophysiology, Neuron doctrine, Ion channels, Action potentials, Synaptic transmission, Synaptic plasticity, Biological basis of learning, Brain damage, Amnesia, Aphasia, Agnosia, Parallel Distributed Processing(PDP), Computational cognitive neuroscience, The appeal of the PDP approach, Biological Basis of Learning, Cajal's synaptic plasticity hypothesis, Long-term potentiation (LTP) and depotentiation (LTD), NMDA receptors and their role in LTP, Synaptic consolidation, Vertical integration, The Problem of representation, Shannon's information theory.

Artificial Intelligence: Turing machines, Physical symbol systems, Symbols and Search Connectionism, Machine Learning**,,** Weak versus strong AI, Subfields, applications, and recent trends in AI, Turing Test revisited, SHRDLU, Heuristic search, General Problem

Solver (GPS), Means-ends analysis.

Cognitive architectures: Tripartite architecture, Integration, ACT-R Architecture Modularity.

Course Learning Outcomes (CLOs):

Upon completion of the course, the students will be able to:

1. identify cognitive science as an interdisciplinary paradigm of study of cross-cutting areas such as Philosophy, Psychology, Neuroscience, Linguistics, Anthropology, and Artificial Intelligence.
2. explain various processes of the mind such as memory and attention, as well as representational and modelling techniques that are used to build computational models of mental processes;
3. acquire basic knowledge of neural networks, linguistic formalism, computing theory, and the brain.
4. apply basic Artificial Intelligence techniques to solve simple problems.

Recommended Books

1. *Bermúdez, J.L., Cognitive Science: An Introduction to the Science of the Mind (2nd Ed,), Cambridge, UK: Cambridge (2014).*
2. *Friedenberg ,J,D, and Silverman,G, Cognitive Science: An Introduction To The Study Of Mind, Sage Publications:, London* (2014)
3. *Thagard, P., Mind: An introduction to Cognitive Science, MIT Press, (2005)*
4. *Thagard, P., (1998) Mind Readings: Introductory Selections on Cognitive Science, MIT Press, Cambridge, Mass,*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UPH064: NANOSCIENCE AND NANOMATERIALS

L	T	P	Cr
2	0	0	2.0

Course Objective:

To introduce the basic concept of Nanoscience and advanced applications of nanotechnology,

Fundamental of Nanoscience: Features of Nanosystem, Free electron theory and its features, Idea of band structures, Density of states in bands, Variation of density of state and band gap with size of crystal,

Quantum Size Effect: Concepts of quantum effects, Schrodinger time independent and time dependent equation, Electron confinement in one-dimensional well and three-dimensional infinite square well, Idea of quantum well structure, Quantum dots and quantum wires,

Nano Materials: Classification of Nano Materials their properties, Basic concept relevant to application, Fullerenes, Nanotubes and nano-wires, Thin films chemical sensors, Gas sensors, Vapour sensors and Bio sensors,

Synthesis and processing: Sol-gel process, Cluster beam evaporation, Ion beam deposition, Chemical bath deposition with capping techniques and ball milling, Cluster assembly and mechanical attrition, Sputtering method, Thermal evaporation, Laser method,

Characterization: Determination of particle size, XRD technique, Photo luminescence, Electron microscopy, Raman spectroscopy, STEM, AFM,

Applications: Photonic crystals, Smart materials, Fuel and solar cells, Opto-electronic devices

Course outcomes:

Upon completion of the course, Students will be able to

1. discriminate between bulk and nano materials,
2. establish the size and shape dependence of Materials' properties,
3. correlate 'quantum confinement' and 'quantum size effect' with physical and chemical properties of nanomaterials,
4. uses top-down and bottom-up methods to synthesize nanoparticles and control their size and shape
5. characterize nanomaterials with various physico-chemical characterization tools and use them in development of modern technologies

Recommended Books:

1. *Booker, R., Boysen, E., Nanotechnology, Wiley India Pvt, Ltd, (2008)*
2. *Rogers, B., Pennathur, S., Adams, J., Nanotechnology, CRS Press (2007)*
3. *Bandyopadhyay, A,K., Nano Materials, New Age Int,, (2007)*
4. *Niemeyer, C. N., and Mirkin, C, A., Nanobiotechnology: Concepts, Applications and Perspectives, Wiley VCH, Weinhein, Germany (2007)*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UMA066: GRAPH THEORY AND APPLICATIONS

L	T	P	Cr
2	0	0	2.0

Course Objective:
The objective of the course is to introduce students with the fundamental concepts in graph Theory, with a sense of some its modern applications. They will be able to use these methods in subsequent courses in the computer, electrical and other engineering,

Introduction: Graph, Finite and infinite graph, incidence and degree, Isolated vertex, Pendent vertex and null graph, Isomorphism, Sub graph, Walks, Paths and circuits, Euler circuit and path, Hamilton path and circuit, Euler formula, Homeomorphic graph, Bipartite graph, Edge connectivity, Computer representation of graph, Digraph.
Tree and Fundamental Circuits: Tree, Distance and center in a tree, Binary tree, Spanning tree, Finding all spanning tree of a graph, Minimum spanning tree.
Graph and Tree Algorithms: Shortest path algorithms, Shortest path between all pairs of vertices, Depth first search and breadth first of a graph, Huffman coding, Cuts set and cut vertices, Warshall's algorithm, topological sorting.
Planar and Dual Graph: Planner graph, Kuratowski's theorem, Representation of planar graph, five-color theorem, Geometric dual.
Coloring of Graphs: Chromatic number, Vertex coloring, Edge coloring, Chromatic partitioning, Chromatic polynomial, covering.
Application of Graphs and Trees: Konigsberg bridge problem, Utilities problem, Electrical network problem, Seating problem, Chinese postman problem, Shortest path problem, Job sequence problem, Travelling salesman problem, Ranking the participant in a tournament, Graph in switching and coding theory, Time table and exam scheduling, Applications of tree and graph in computer science.

Course Learning Outcomes (CLOs):
Upon completion of the course, the students will be able to:

1) understand the basic concepts of graphs, directed graphs, and weighted graphs and able to present a graph by matrices.
2) understand the properties of trees and able to find a minimal spanning tree for a given weighted graph.
3) understand Eulerian and Hamiltonian graphs.
4) apply shortest path algorithm to solve Chinese Postman Problem .
5) apply the knowledge of graphs to solve the real life problem.

Recommended Books

1. *Deo, N., Graph Theory with Application to Engineering with Computer Science, PHI, New Delhi (2007)*
2. *West, D. B., Introduction to Graph Theory, Pearson Education, London (2008)*
3. *Bondy, J. A. and Murty, U.S.R., Graph Theory with Applications, North Holland Publication, London (2000)*

4. *Rosen, K. H., Discrete Mathematics and its Applications, Tata-McGraw Hill, New Delhi (2007)*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UMA070: ADVANCED NUMERICAL METHODS

L	T	P	Cr
2	0	0	2.0

Course Objectives:
The main objective of this course is to motivate the students to understand and learn various advanced numerical techniques to solve mathematical problems governing various engineering and physical problems.

Non-Linear Equations: Methods for multiple roots, Muller's, Iteration and Newton-Raphson method for non-linear system of equations and Newton-Raphson method for complex roots.
Polynomial Equations: Descartes' rule of sign, Birge-vieta, Giraffe's methods.
System of Linear Equations: Cholesky and Partition methods, SOR method with optimal relaxation parameters.
Eigen-Values and Eigen-Vectors: Similarity transformations, Gerschgorin's bound(s) on eigenvalues, Given's and Rutishauser methods.
Interpolation and Approximation: Cubic and B – Spline and bivariate interpolation, Least squares approximations, Gram-Schmidt orthogonalisation process and approximation by orthogonal polynomial, Legendre and Chebyshev polynomials and approximation.
Differentiation and Integration: Differentiation and integration using cubic splines, Romberg integration and multiple integrals.
Ordinary differential Equations: Milne's, Adams-Moulton and Adam's Bashforth methods with their convergence and stability, Shooting and finite difference methods for second order boundary value problems.

Course Learning Outcomes:
Upon completion of this course, the students will be able to**:**

1) find multiple roots of equation and apply Newton -Raphson's method to obtain complex roots as well solution of system of non - linear equations.
2) learn how to obtain numerical solution of polynomial equations using Birge - Vitae and Giraffe's methods.
3) apply Cholesky, Partition and SOR methods to solve system of linear equations.
4) understand how to approximate the functions using Spline, B- Spline, least square .approximations
5) learn how to solve definite integrals by using cubic spline, Romberg and initial value problems and boundary value problems numerically.

Recommended Books

1) *Gerald, C.F. and Wheatley, P.O., Applied Numerical Analysis, Pearson Education (2008) 7th ed.*
2) *Gupta, S.R., Elements of Numerical Analysis, MacMillan India (2009).*
1) *Atkinson, K.E., An introduction to Numerical Analysis, John Wiley (2004) 2nd ed.*
2) *S.D. Conte, S.D. and Carl D. Boor, Elementary Numerical Analysis: An Algorithmic Approach, Tata McGraw Hill (2005).*

3) *Jain M. K., Iyengar. S.R.K. and Jain, R.K. Numerical Methods for Scientific and Engineering Computation, New Age International (2008) 5th ed.*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55

UBT509: BIOLOGY FOR ENGINEERS

L	T	P	Cr
2	0	0	2.0

Course Objective: To learn about living world and basic functioning of biological systems. The course encompasses understanding of origin of life, its evolution and some of its central characteristics. It also aims to familiarize engineering students to some of the intricate biological phenomena and mechanisms.

Characteristics of life: Living versus non-living organisms, origin of life, theory of evolution, diversity of life, classification of life into animals, plants, fungi, protists, archea and bacteria. Phylogenetics and its relationship with evolution.

Introduction to biological systems: Cell as basic unit of life, cellular organelles and their functions, important biomacromolecules (carbohydrates, lipids, proteins and nucleic acids) and their properties.

Cell membrane: Membrane structure, selective permeability, transport across cell membrane, active and passive transport, membrane proteins, type of transport proteins, channels and pumps, examples of membrane transport in cell physiology.

Classical and molecular genetics: Heredity and laws of genetics, genetic material and genetic information, Structure and properties of DNA, central dogma, replication of genetic information, universal codon system, encoding of genetic information via transcription and translation.

Course Learning Outcomes (CLOs):

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

1. Explain the characteristic features of living-systems and differentiate them from non-living systems
2. Broaden the application of engineering knowledge of their branch by applying concepts of living systems.
3. Demonstrate familiarity with special properties of biological macromolecules
4. Upgrade their understanding about biological systems by drawing parallel with thermodynamics system and develop interface between an engineering specialization and living systems.
5. Design engineering products inspired by living creatures.
6. Plan application of computational tools in bioinformatics.

Recommended Books:

1. *Nelson, D.L., Cox, M.M., Lehninger: Principles of Biochemistry, WH Freeman (2008) 5th ed.*
2. *Dhami, P.S., Srivastava, H.N. Chopra, G., A Textbook of Biology, Pradeep Publications (2008).*
3. *Das, H.K., Textbook of Biotechnology, John Wiley & Sons (2004) 3rd Edition.*
4. *Gardner, E.J., Simmons, M., Peter, S.D., Principles of Genetics, John Wiley & Sons (2008)*
5. *Albert, B., Essential Cell Biology, Taylor & Francis, London (2009)*

Evaluation Scheme:

Sr. No.	Evaluation Elements	Weightage (%)
1	MST	45
2	EST	55