INFORMATION TECHNOLOGY DEPARTMENT

<table>
<tr><td rowspan="2">Subject: Data Structures (Code: ITT201)</td><td rowspan="2" colspan="2">Year & Semester: B. Tech Information Technology Engineering
2nd Year & 3rd Semester</td><td colspan="3">Total Course Credit: 4</td></tr>
<tr><td>L</td><td>T</td><td>P</td></tr>
<tr><td rowspan="2">Evaluation Policy</td><td rowspan="2"></td><td>3</td><td>1</td><td>0</td></tr>
<tr><td>Mid-Term</td><td>Class Assessment</td><td colspan="3">End-Term</td></tr>
<tr><td>30 Marks</td><td>10 Marks</td><td colspan="3">60 Marks</td></tr>
</table>

Course Outcomes (COs):

CO1: Understand basic data structures such as arrays, strings, and linked lists.

CO2: Study linear data structures such as stacks and queues and understand their difference.

CO3: Describe the hash function and concepts of collision and its resolution methods.

CO4: Understand the concept of memory management.

CO5: Study tree, heap and graphs along with their basic operations.

CO6: Study different techniques for solving problems like sorting and searching

Syllabus:

Introduction: Basic concept of data, structures and pointers.

Arrays: Representation, implementation, polynomial representation. Limitations.

Strings: Representation, String operations, Implementing String.h library functions.

Linked List: Static and dynamic implementation. Single, double, circular, multiple linked lists.

Stacks: Recursion and Stacks. Static and dynamic implementation. Expression evaluation. Infix, postfix expressions, multiple stacks.

Queues: Static and dynamic implementation, circular queues, and implementation.

Hash Tables: Hash tables implementation. Hashing techniques, single, double.

Storage Management: Memory Management techniques, garbage collection.

Trees: Binary trees, binary search trees, static and dynamic implementation. Tree operations, insert, delete, and search.

Heaps: Implementation, sorting etc.

Sorting and Searching: Different sorting techniques. Insertion sort, selection sort, bubble sort, radix sort, quick sort, merge sort, heap sort.

Graphs: Representation of graphs, BFS, DFS sort. Graph Algorithms.

Books Recommended:

1. Data Structures by Rajni Jindal

2. Data Structures - Schaum‘s Series

3. Data Structures by Knuth

4. Data Structures by Farouzan

5. Data Structures using C and C++ by Langsam, Augestern, Tanenbaum.

Subject: Signals & Systems (Code: ITT202)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester		Total Course Credit: 3		
			L	T	P
			2	1	0
Evaluation Policy	Mid-Term	Class Assessment	End-Term		
	30 Marks	10 Marks	60 Marks		

Course Outcomes (COs):

CO1. Understand the basic classification, properties and operations on signals

CO2. Understand the basic classification, properties of systems and LTI systems.

CO3. Finding the Fourier and inverse transform of signals and its properties

CO4. Finding the Laplace and Inverse transform of signals and its properties.

CO5. To understand use of probability distribution function and density for signals and systems

CO6. Using autocorrelation and cross correlation functions and noise in LTI systems

Syllabus:

1. Introduction to signals:

Classification of signals; Deterministic and non-deterministic, periodic and aperiodic, even and odd signals, energy and power signals, elementary signals; exponential, sinusoidal, impulse, step, ramp, pulse, square wave signals. Time shifting, time scaling and time-inversions of signals

2. Linear Time invariant systems

Continuous time system, basic system properties like causality, time invariance, stability, linearity, memory, order of system, interconnection of systems, Linear time invariant systems, characterization, unit impulse response , convolution, properties of LTI systems, linear constant co-efficient differential equations and system description.

3. Fourier analysis of signals and systems

Fourier series of periodic signals and its properties, Fourier transform of aperiodic signals andits properties, fourier transform of periodic signals, convolution in time and frequency domain, energy and signals, parsevals theorem, energy spectral density and its properties, Transfer function of LTI system.

4. The Laplace Transform

Definition, relation between Laplace and Fourier transforms, region of convergence, properties of Laplace transform, initial and final value theorems, convolution, transfer function of LTI system, concept of poles and zeroes, stability criteria.

5. Random variable theory and random signals

Probability, conditional probability, statistical independence, random variables, discrete and continuous random variables, probability distribution and probability density functions, statistical averages of random variables. Some important density functions.

6. Random processes and characterization

Ensemble and time averages, stationary and non-stationary random process, wide sense stationary random process, autocorrelation and cross-correlation functions, response of LTI systems to random inputs, noise and its types, white noise, signal to noise ratio of LTI systems.

Text Books:

1. Signals and Systems by Zieman, Tranter, Fannin
2. Signals and Systems by Sanjay Sharma

Reference Books:

1. Signals and Systems by A Populis
2. Random processes and Systems by A Populis
3. Signals and Systems by S. Hykin

Subject: Software Engineering (Code: ITT203)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester		Total Course Credit: 3		
			L	T	P
			3	0	0
Evaluation Policy	Mid-Term	Class Assessment	End-Term		
	30 Marks	10 Marks	60 Marks		

Course Outcomes (COs):

CO1	Students will know classical and evolving software engineering methods, can select and tailor appropriate methods for projects, and can apply them as both team members and managers to achieve project goals.
CO2	Ability to effectively apply software engineering practice over the entire system lifecycle. This includes requirements engineering, analysis, prototyping, design, implementation, testing, maintenance activities and management of risks involved in software and embedded systems.
CO3	To make students proficient in effective written and oral communication skills so that they can prepare and publish the necessary documents required throughout the project lifecycle. It also includes effectively contributing to project discussions, presentations, and reviews.
CO4	To make students knowledgeable of the ethics, professionalism, and cultural diversity in the work environment and develop an awareness of the role and responsibilities of the professional software Engineers and Understanding software testing approaches such as unit testing and integration testing along with the need for lifelong learning and readily adapting to new software engineering environments.

Syllabus:

Unit I - Software Process:

The Evolving role of Software , Defining Software, Software Myths, Legacy software, A generic view of process, A layered Technology, Process Framework, Capability Maturity Model Integration (CMMI), Process Assessment, Personal and Team Process Models, Product and Process, Process Models – Build and fix model, The Waterfall Model, Incremental Process

Model, RAD Model, Evolutionary Process Models, Unified Process, Agile Methodology, SCRUM Approach.

Unit II - Requirement Engineering:

Software Engineering Practice, Requirements Engineering tasks, Types of requirements, Feasibility studies, initiating the requirements Engineering Process, Eliciting Requirements, Developing Use cases, Requirement Analysis, Documentation and validation, Building the Analysis Model, Elements of the Analysis Model

Unit III - Analysis Modeling And Project Planning:

Requirements Analysis, Analysis Modeling approaches, data modeling concepts, Object oriented Analysis, Scenario based modeling, Flow oriented Modeling, Class based modeling, creating a behavior model. Planning: Size estimation, Cost estimation, COCOMO, Software risk management.

Unit IV - Design & Implementation:

Design Engineering, Design Concepts, Modularity, Strategy of Design, Function oriented Design, Architectural Design, Detailed Design, Design process, Design Quality, Design model, User interface Design, Implementation, issues in implementation. Software metrics, SCM.

Unit V - Testing & Maintenance:

Testing strategies, Testing Tactics and terminologies, functional testing, structural testing, levels of testing, validation testing, system testing, Art of debugging. Software maintenance, maintenance models, Regression testing, Reverse Engineering, Re-Engineering, evolution, Quality Management, Process Improvement, Risk Management.

Text Books:

1. Ian Sommerville, “Software Engineering”, Pearson Education.
2. Software Engineering – A Practitioner’s Approach, Roger S. Pressman, McGraw Hill.
3. Software Engineering, K.K. Aggarwal, Yogesh Singh, New Age International Publishers.

Reference Books:

1. Richard Fairley, "Software Engineering Concepts", McGraw Hill.

2. Stephan Schach, “Software Engineering”, Tata McGraw Hill.
3. Pfleeger and Lawrence,“Software Engineering: Theory and Practice”, Pearson Education.

Subject: Discrete Mathematics & Graph Theory (Code: ITT204)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester		Total Course Credit: 4		
			L	T	P
			3	1	0
Evaluation Policy	Mid-Term	Class Assessment	End-Term		
	30 Marks	10 Marks	60 Marks		

Course Outcomes (COs):

CO1	Understand discrete structures such as sets, relations, and lattices.
CO2	Study the basic operations of Propositional logic and Boolean Algebra.
CO3	Analyse and study various proof techniques.
CO4	Understand basics of Graph theory and how it can be used to visualize and simplify problems.
CO5	Understand the various properties of algebraic systems like Rings, Monoids and Groups.

Syllabus:

Unit I - Sets and Relations:

Sets, Relations, Representation of Relations, Composition of Relations, Partitions, Equivalence Relations. Ordered sets and Lattices: Ordered sets, Diagram of Partially ordered sets,

Supremum and Infimum, well ordered sets, Lattices, Bounded and complemented lattice, Distributive Lattice.

Unit II - Propositional Calculus:

Statements, Basic operations, Truth value of compound statements, Algebra of Propositions, Tautologies and contradiction, Conditional and Bi-conditional statements, logical implications, logical equivalence, predicates, Universal and existential quantifiers.Logic gates, Boolean Algebra, Postulates of Boolean Algebra; Theorems of Boolean Algebra, Sum of products and product of sums Simplification.

Unit III - Proof techniques:

Notions of implication, converse, inverse, contrapositive, negation, and contradiction; the structure of formal proofs; direct proofs; proof by counterexample; proof by contraposition; proof by contradiction; mathematical induction.

Unit IV - Graph Theory:

Graphs and Multi-graphs, Degree of a vertex, Paths connectivity, Cut points Bridges, Walks, paths, cycles, connected graphs, Bipartite, Regular,Planar and connected graphs,Euler graphs, Euler's theorem, Hamiltonian path and circuits, Graph coloring, chromatic number, isomorphism and Homomorphism of graphs,Konigsberg seven bridge problem, Shortest path. Trees, properties of trees, pendant vertices in trees, Degree sequences in trees, Necessary and sufficient conditions for a sequence to be a degree sequence of a tree.

Unit V - Group Theory:

Groups, semigroup, infinite group, Finite group, order of a group, Abelian group, subgroup, Lagrange's Theorem, Cosets, Normal Subgroups, order of an element of a group, cyclic group. Rings , Homomorphism and Isomerphism of rings.

Books Recommended

1. C. L. Liu : Elements of Discrete Mathematics, 2nd Ed.Tata Mc-Graw Hill.

2. Kolman, Busby and Ross : Discrete Mathematical Structers, 6th Ed.PHI (2009).

3. NarsinghDeo : Graph Theory with Applications to Engineering and Computer Sciences, PHI.

4. Murry R. Spiegel: Discrete Mathematics (Schaums Outline series) Tata McGraw Hill (2009).

5. K. Bogart, S. Drysdale, C. Stein. Discrete Math for Computer Science Students.

Reference Books

1. Kenneth H. Rosen:Discrete Mathematics and its applications,Tata McGraw Hill (2003).

2. K.R Parthasarty : basic Graph Theory, Tata Mc-Graw Hill

Subject: Object Oriented Programming (Code: CST201)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 4**		
		L	T	P
		3	1	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

· To explore the principles of Object Oriented Programming (OOP).

· To understand object-oriented concepts such as data abstraction, encapsulation, inheritance, dynamic binding, and polymorphism.

· To use the object-oriented paradigm in program design.

· To lay a foundation for advanced programming.

· Provide programming insight using OOP constructs

Learning Outcomes

On completion of the course, student will be able to:

- Analyze the strengths of object oriented programming
- Design and apply OOP principles for effective programming
- Develop programming application using object oriented programming language C++
- Percept the utility and applicability of OOP.

Course Outline / Content	
Unit	**Topics**
1.	**Classes and Objects:** Need of Object-Oriented Programming (OOP), Object Oriented Programming Paradigm, Basic Concepts of Object-Oriented Programming, Benefits of OOP, C++ as object oriented programming language. C++ Programming- C++ programming Basics, Data Types, Structures, Enumerations, control structures, Arrays and Strings, Class, Object, class and data abstraction, class scope and accessing class members, separating interface from implementation, controlling access to members. Functions- Function, function prototype, accessing function and utility function, Constructors and destructors, Copy Constructor, Objects and Memory requirements, Static Class members, data abstraction and information hiding, inline function.

2.	**Polymorphism and Inheritance:** Operator Overloading- concept of overloading, operator overloading, Overloading Unary Operators, Overloading Binary Operators, Data Conversion, Type casting (implicit and explicit), Pitfalls of Operator Overloading and Conversion, Keywords explicit and mutable. Inheritance- Base Class and derived Class, protected members, relationship between base Class and derived Class, Constructor and destructor in Derived Class, Overriding Member Functions, Class Hierarchies, Inheritance, Public and Private Inheritance, Levels of Inheritance, Multiple Inheritance, Ambiguity in Multiple Inheritance, Aggregation, Classes Within Classes. Polymorphism- concept, relationship among objects in inheritance hierarchy, abstract classes, polymorphism.
3.	**Virtual Functions:** Virtual Functions- Pointers- indirection Operators, Memory Management: new and delete, Pointers to Objects, A Linked List Example, accessing Arrays using pointers, Function pointers, Pointers to Pointers, A Parsing Example, Debugging Pointers, Dynamic Pointers, smart pointers, shared pointers, Case Study : Design of Horse Race Simulation. Virtual Function- Friend Functions, Static Functions, Assignment and Copy Initialization, this Pointer, virtual function, dynamic binding, Virtual destructor.
4.	**Templates and Exception handling:** Templates- function templates, Function overloading, overloading Function templates, class templates, class template and Non-type parameters, template and inheritance, template and friends Generic Functions, Applying Generic Function, Generic Classes, The typename and export keywords, The Power of Templates. Exception Handling- Fundamentals, other error handling techniques, simple exception handlingDivide by Zero, rethrowing an exception, exception specifications, processing unexpected exceptions, stack unwinding, constructor, destructor and exception handling, exception and inheritance.
5.	**Files and Streams:** Data hierarchy, Stream and files, Stream Classes, Stream Errors, Disk File I/O with Streams, File Pointers, and Error Handling in File I/O, File I/O with Member Functions, Overloading the Extraction and Insertion Operators, memory as a Stream Object, Command-Line Arguments, Printer output, Early vs. Late Binding.

1.	**Standard Template Library:** Standard Template Library, Overview of Standard Template Library, Containers, Algorithms, Iterators, Other STL Elements, Container Classes, General Theory of Operation, Vectors.

Books Recommended

Text Books	1. Robert Lafore, "Object Oriented Programming in Turbo C++", Galgotia Publications, 2. Balagurusamy, "Object Oriented programming with C++", Tata McGraw Hill.
References	1. BjarneStrustrup, "The C++ programming Language", Addison Wesley, 2. Booch, "Object Oriented Analysis and Design with Applications, Addison Wesley. 3. Chair H. Pappas & William H. Murray, "The Complete Reference Visual C++", TMH.

Subject: Electronics (Code: ECT207)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 4**		
		L	T	P
		3	1	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

· To get basic idea about types, specification and common values of passive components.

· To familiarize the working and characteristics of diodes, transistors, MOSFETS and some measuring instruments.

· To understand working of diodes in circuits and in rectifiers.

Learning Outcomes

Student can identify the active and passive electronic components. Student can setup simple circuits using diodes and transistors. Student will get fundamental idea about basic communication systems and entertainment electronics.

Course Outline / Content	
Unit	**Topics**
1.	**Semiconductors:** Insulators, semiconductors and metals, Mobility and conductivity, Intrinsic and extrinsic semiconductors, Charge Densities in Semiconductors, Mass action Law, Current Components in Semiconductors, The Continuity Equation, Injected minority Charge Carrier, Hall effect.
2.	**PN Junction Diode:** Characteristic and analysis, Types of diodes – Zener diodes, Photodiodes, Light emitting diodes (LED‟s), Varactor diodes and tunnel diodes. Rectifiers and filter circuit: Half wave, full wave and Bridge rectifier circuits and their analysis, L, C and Pi filters, Basic regulator supply using zener diode, Clipping and clamping circuits.

3.	**Transistors:** Construction and characteristics of bipolar junction, transistors (BJT‟s)-Comm. Base, Comm. emitter, Comm. Collector configuration. Transistor at low frequencies – small signal low frequency transistor model (hparameters). Analysis of transistor amplifier circuit using h-parameters. Transistor biasing and bias stabilization: the operating point, stability factor, analysis of fixed base bias, collector to base bias, Emitter resistance bias circuit and self bias circuit. Bias compensation techniques.
4.	**Field Effect Transistor:** Construction and characteristics of JFET, JFET biasing circuit, JFET amplifier, MOSFET construction and characteristics.
5.	**Amplifiers And Oscillators:** Classification of amplifiers, concept of feedback, general characteristics of feedback amplifiers, Single stage RC coupled amplifier. Oscillators – Criterion for Oscillation, type of oscillators: Hartley oscillator, Colpitt Oscillator, RC Phase shift oscillator, Crystal oscillator.
6.	**Operational Amplifiers:** Introduction to Op-amp, Inverting and non-inverting configuration, Applications – adder, subtractor, integrator, differentiator.

Books Recommended

Text Books	1. Bhargava N. N., D C Kulshreshtha and S C Gupta, “Basic Electronics & Linear Circuits”, Tata McGraw Hill, 2/e, 2013. 2. Electronics Devices and Circuit Theory by R. Boylestad, Pearson.
References	1. Bell, D. A., Electronic Devices and Circuits, Oxford University Press. 2. Boylested, R. L. and Nashelsky, L., Electronic Devices and Circuit Theory, Pearson Education. 3. Frenzel, L. E., Principles of Electronic Communication Systems, McGraw Hill.

Subject: Object Oriented Programming Lab (Code: CSL203)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester	Total Course Credit: 1		
		L	T	P
		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

· The student should be able to explain the fundamental properties of the C++ language.

· The student should be able to combine the elements of the C++ language in developing structured programs.

· The student should be able to demonstrate the skills necessary to correctly compile, debug, and test programs in C++.

Learning Outcomes

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

· Apply C++ features to program design and implementation

· Explain object-oriented concepts and describe how they are supported by C++

· Use C++ to demonstrate practical experience in developing object-oriented solutions

· Analyse a problem description and design and build object-oriented software using good coding practices and techniques

· Use common software patterns in object-oriented design and recognise their applicability to other software development contexts.

Course Outline / Content	
Unit	**Topics**
1.	Function overloading and default arguments in C++
2.	Simple class design in C++, namespaces, object creation
3.	Class design in C++ using dynamic memory allocation
4.	Constructors and destructors
5.	Operator overloading and friend functions
6.	Overloading assignment operator and type conversions

7.	Inheritance, run time polymorphism and virtual functions
8.	Template design in C++
9.	Interface and abstract classes
10.	Exception handling
11.	File handling in C++

Books Recommended

Text Books	1. Robert Lafore, "Object Oriented Programming in Turbo C++", Galgotia Publications, 2. Balagurusamy, „Object Oriented programming with C++", Tata McGraw Hill.
References	1. BjarneStrustrup, "The C++ programming Language", Addison Wesley, 2. Booch, "Object Oriented Analysis and Design with Applications, Addison Wesley. 3. Chair H. Pappas & William H. Murray, "The Complete Reference Visual C++", TMH.

Subject: Electronics Lab (Code: ECL208)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester	Total Course Credit: 1		
		L	T	P
		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

· To familiarize with the electronic components and basic electronic instruments.

· To enable the students to understand the behaviour of semi conductor devices based on experimentation.

Learning Outcomes

· To make familiar with PCB design and various processes involved.

· Ability to understand and analyse, linear and digital electronic circuits.

Course Outline / Content	
Unit	**Topics**
1.	Characteristics of Semi conductor diode and Zener diode

2.	Characteristics of a NPN Transistor under common emitter, common collector and common base configurations
3.	Characteristics of JFET (Draw the equivalent circuit)
4.	Characteristics of UJT and generation of saw tooth waveforms
5.	Design and Frequency response characteristics of a common emitter amplifier
6.	Design and testing of RC phase shift, LC oscillators
1.	Single phase half-wave and full wave rectifiers
8.	a) To assemble a half wave and a full wave rectifier and to study their performance. b) To suppress the ripple using RC filter.
9.	To assemble and observe the performance of clipping and clamping circuits.
10.	Design and realize Inverting and Non-inverting amplifier using 741 Op-amp.

Books Recommended

Text Books	1. Bhargava N. N., D C Kulshreshtha and S C Gupta, "Basic Electronics & Linear Circuits", Tata McGraw Hill, 2/e, 2013 . 2. Electronics Devices and Circuit Theory by R. Boylestad, Pearson.
References	1. Bell, D. A., Electronic Devices and Circuits, Oxford University Press. 2. Boylested, R. L. and Nashelsky, L., Electronic Devices and Circuit Theory, Pearson Education

	3. Frenzel, L. E., Principles of Electronic Communication Systems, McGraw Hill.

Subject: Data Structures Lab (Code: ITL205)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 1**		
		L	T	P
		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Outcomes (COs):

CO1 Implement and understand linked lists for polynomial manipulation.

CO2 Study stack data structure and use it for expression parsing (prefix, infix, postfix).

CO3 Implement circular queues for producer-consumer problem simulation.

CO4 Implement search trees and priority queues using heaps.

CO5 Implement various hashing techniques.

CO6 Study and implement various searching and sorting techniques.

C07 Implement various Graph Algorithms

Syllabus:

Basic concepts of data, linear lists, strings, arrays and orthogonal lists, representation of trees & graphs, storage systems, Arrays, Recursion, Stacks, Queues, Linked lists, Binary trees, General Trees, Tree Traversal, Symbol Table and Searching Techniques, Sorting Techniques,

Graphs:

1. Implement singly and doubly linked lists.
2. Represent a polynomial as a linked list and write functions for polynomial addition.
3. Implement stack and use it to convert infix to postfix expression
4. Implement array-based circular queue and use it to simulate a producer-consumer problem.
5. Implement an expression tree. Produce its pre-order, in-order, and post-order traversals.
6. Implement a binary search tree.
7. Implement priority queue using heaps
9. Implement hashing techniques
10. Implement various sorting techniques as taught in class.
11. Implement Dijkstra's algorithm using priority queues.
12. Implement Prim's and Kruskal's algorithms

Subject: Operating Systems (Code: ITT250)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester	Total Course Credit: 4		
		L	T	P
		3	1	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Outcomes (COs):

At the end of the course students will be able to understand the:

CO1: Functions of operating System.

CO2: Operating System processes, management and coordination.

CO3: Interprocess Communication and process control.

CO4: Deadlocks detection, prevention and avoidance mechanisms.

CO5: Process Scheduling Algorithms.

CO6: Memory and I/O Device Management.

Syllabus:

UNIT I

INTRODUCTION:

Computer System Overview-Basic Elements, Instruction Execution, Operating system functions and structure, Interrupts, Memory Hierarchy, Cache Memory, Direct Memory Access, Multiprocessor and Multicore Organization. Operating system overview-objectives and functions, Evolution of Operating System, Distributed OS.

UNIT II

PROCESS MANAGEMENT AND COORDINATION:

Process concept, Process States, Process Description and Process Control, Interprocess Communication, Processes and Threads, Types of Threads, Multicore and Multithreading,

UNIT III

CONCURRENCY AND SCHEDULING:

Principles of Concurrency - Mutual Exclusion, Semaphores, Monitors, Readers/Writers problem. Deadlocks – prevention- avoidance – detection, Scheduling- Types of Scheduling – Scheduling algorithms.

UNIT IV

MEMORY MANAGEMENT:

Memory management requirements, Partitioning, Paging and Segmentation, Virtual memory - Hardware and control structures, operating system software, Linux memory management, Windows memory management. Virtual memory management.

UNIT V

INPUT/OUTPUT AND FILE SYSTEMS:

I/O management and disk scheduling – I/O devices, organization of I/O functions; OS design issues, I/O buffering, disk scheduling, Disk cache. File management – Organization, Directories, File sharing, and Record blocking, secondary storage management.

Text Books:

1. Silberschatz, Peter Galvin, Greg gagne "Operating System Principles".
2. William Stallings, "Operating Systems – internals and design principles", Prentice Hall.

Reference Books:

1. Andrew S. Tannenbaum & Albert S. Woodhull, "Operating System Design and Implementation", Prentice Hall.
2. Andrew S. Tannenbaum, "Modern Operating Systems", Prentice Hall.
3. Gary J.Nutt, "Operating Systems", Pearson/Addison Wesley.
4. Pramod Chandra P.Bhatt, "An Introduction to Operating Systems Concepts and Practice".

Subject: Database Management System (Code: ITT251)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 4**		
		L	T	P
		3	1	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Outcomes (COs):

CO1: Identify the basic concepts and various data model used in Database design.

CO2: Apply relational database theory and be able to describe relational algebra expression, tuple and domain relation expression for queries and SQL for implementing the queries.

CO3: Recognize and identify the use of normalization and functional dependency.

CO4: Apply and relate the concept of transaction, concurrency control and recovery in database.

CO5: Recognize / identify the purpose of query processing and optimization, indexing and hashing technique used in database design.

Syllabus:

UNIT I

INTRODUCTION:

Introduction to database management, data abstraction and system structure, Purpose of database system , uses of database approach, database applications, Views of data, Database languages, Database system – Concepts and architecture, Database users and administrator, database types.

UNIT II

DATA MODELLING:

Data models definition and types, Entity- Relationship Model (E-R Model), E-R diagrams, entity set, relationship sets, mapping, cardinalities. Introduction to relational databases, The relational model - Keys, Relational algebra – Domain relational calculus – Tuple relational calculus – Fundamental operations – Additional operations – SQL fundamentals, Views, Introduction to distributed databases and client/server databases.

UNIT III

DATABASE DESIGN:

Relational database design, Functional dependencies, Non-loss decomposition, First, Second, Third Normal Forms – Dependency Preservation – Boyce/Codd Normal Form, Multi-Valued Dependencies and higher normal Forms.

UNIT IV

TRANSACTIONS:

Transaction Concepts, Transaction Recovery, ACID Properties, System Recovery, Media recovery, Two phase commit, Save points, SQL facilities for recovery, Concurrency, Need for concurrency, Locking protocols - Two phase locking, Intent locking, Deadlock, Serializability, Recovery isolation levels, SQL facilities for concurrency**.**

UNIT V

IMPLEMENTATION TECHNIQUES:

Overview of physical storage media – Magnetic disks, Tertiary storage, File organization – Organization of records in files, Indexing and hashing, ordered indices, B trees index files, Static hashing, dynamic hashing, RAID organization and levels. Data warehouse and data mining- basic concepts and overview.

Text Books:

1. R. and Navathe, S.B., “Fundamentals of Database Systems”, Pearson Education.

Reference Books:

1. Abraham, H. and Sudershan, S., “Database System Concepts”,McGraw-Hill.Elmasri.
2. Ramakrishnan, R. and Gekhre, J., “Database Management Systems”, McGraw-Hill.

Subject: Digital Electronics & Logic Design (Code: ECT251)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester	Total Course Credit: 4		
		L	T	P
		3	1	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

To study number systems, simplification and implementation of digital functions, design & analysis of various combinational and sequential circuits, memory organization & its types and also understand basics of VHDL programming.

Learning Outcomes

The student will be able to:

- Use number systems, binary addition and subtraction.
- Understand the different switching algebra theorems and apply them for logic functions.
- Use the Karnaugh map for reduction of logic functions.
- Design the combinational circuits.
- Design the sequential circuits.
- Derive the state-machine analysis or synthesis.

Course Outline / Content	
Unit	**Topics**
1.	**Binary Systems:** Number Systems (binary, octal, hexadecimal), conversion from one system to another, addition and subtraction using different number systems, complements and codes.
2.	**Boolean algebra & Logic Gates:** Basic Definitions, Theorems and Properties of Boolean Algebra, Boolean functions, Canonical and Standard Forms, Logic Operations & Gates
3.	**Simplification of Boolean Functions:** K-Map Method and Tabulation Method (2, 3, 4, 5 variables)
4.	**Combinational Logic:** Design Procedure, Logic gates and Arithmetic Circuits
5.	**Combinational Logic with MSI & LSI:** Adder, Subtractor, Encoders, Decoders,Multiplexers,De-multiplexers ,ROMs, PLA‘s
6.	**Sequential Logic:** Moore and Mealy Machine Design Procedure state machine as a sequential controller,Flip-Flops (FF), Triggering, Analysis, State Reduction &Assignment. FF Excitation Tables, ASM Charts, Design Procedure, Design of Counters, Design with State Equations.
7.	**Registers, Counters:** Shift Registers, Synchronous and Asynchronous Counters **Data Converters:** ADC, DAC and their types.
8.	**VHDL Programming:** Introduction, Code Structure, Data Types Operators &Attributes, Concurrent Code, Sequential Code, Signals & Variables, Basic Circuit Designs.

Text Books

1. Digital Logic & Computer Design by M Morris Mano
2. Digital Electronics by Gupta &Singhal
3. Circuit Design with VHDL, V A Pedroni.

References

1. Digital principles and applications by A. P. Malvino
2. Switching Circuits by Marcus

Subject: Communication System (Code: ECT253)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 4**		
		L	T	P
		3	1	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

This course has been designed to familiarize students with the fundamentals of design and analysis of both analog and digital communications systems. In the first half of this course, "how a communications system works from signal modulation and transmission point of view" is discussed and in the second half the course "the behavior of the communications systems in the presence of noise" is discussed. In the last two weeks of this course, two most important areas of communications i.e., wireless and optical fiber communications are introduced to students to further motivate their interest in future communications courses and ultimately a career in communications industry.

Learning Outcomes

Upon completion of this course, students will have the ability to:

- Understand operation of digital and analog communication systems
- Obtain knowledge of theoretical principles of communication systems and statistical
- Properties of noise so to be able to apply them to engineering systems (a,c,e) should develop a knowledge and understanding of advanced communication.

Course Outline / Content	
Unit	**Topics**

1.	Spread Spectrum Communication: Direct sequence and frequency hopped spread spectrum, spreading sequences and their correlation functions, Acquisition and tracking of spread spectrum signals Code Division Multiple Access (CDMA): DS-CDMA on AWGN channels, DS-CDMA on frequency selective fading channels, Performance analysis of cellular DS-CDMA, Capacity estimation, Power control effect of imperfect power control on DS-CDMA performance, Soft Hand offs, Spreading/coding tradeoffs, multi carrier CDMA, IS95A CDMA systems, 3rd Generation CDMA systems, Multi user detection, Optimum receivers, SIC, PIC receivers and performance.
2.	Networks & Services: Network Transmission System Design Services, Characterization of networks & teleservices, The Telephone Network – Past, Present & Future, and Network issues.
3.	Data Communication Networks: Basic principles of data communication – synchronous and asynchronous transmission – digital data transmission formats NRZ, RZ, AMI, ASI & Manchester coding, Error correcting codes, Hamming codes, Orthogonal codes, Switching – Circuit switching, Message switching, Packet switching, Standard communication interface multipliers and concentrators, Protocols (BOP-COP – standard networks and standards, OSI, (D) ARPANET, NICNET, SNA, SELS etc. Lan types of LAN – WAN, Digital telephony, Basic principle of ISDN – E Mail – Voice mail.
4.	Transmission Principles: Transmission aspects, Signals and Impairments, Digital Speech Transmission Digitisation of Speech & Audio.
5.	Teletraffic: Digital Networks, Network Synchronization, Multiplexing – Digital Hierarchies, Synchronous Digital Hierarchy, Digital Switching, Signaling, Introduction to Teletraffic.
6.	ISDN & ATM: Integrated Services Digital Network – ISDN, Broadband ISDN & ATM, Broadband Access Networks, Optical Networks. Network Aspects: Intelligent Network, Network Management, and Introduction to Network management softwares.

Text Books

1. Andrew J Viterbi, “CDMA Principles of spread spectrum communications”, Addition Wesley, (1995).

2. J S Lee and L E Miller, “CDMA systems engineering handbook”, Artech House, (1998).

References

1. Marvin K Simon, Jim K Omura, Robert A Scholtz, BaryKlevit, “Spread Spectrum Communications”, (1995).

2. Sergio Verdu, “Multiuser Detection”, Cambridge University Press, (1998).

3. Andrew S Tanenbaum, “Computer Networks”, Prentice Hall of India.

Subject: Control System (Code: EET258)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 3**		
		L	T	P
		3	0	0
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

- To develop an understanding of principles and applications of control systems in everyday life.
- To understand the basic concepts of block diagram reduction, time domain analysis solutions to time invariant systems.
- To develop an understanding of different aspects of stability analysis of systems in frequency domain and time domain.
- Design controllers to meet specifications.

Learning Outcomes

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

- Should have knowledge on open loop and closed loop control systems, concept of feedback in control systems etc.
- Should be able to apply the conceptual things to real-world electrical and electronics problems and applications.
- Test a linear system for stability by determining the system's pole locations.
- Test a linear system for controllability and observability.
- Should be able to develop and run a computer simulation of a control system using MATLAB.

Course Outline / Content

Unit	Topics
1.	**Introduction:** Concepts of Control Systems- Open Loop and closed loop control systems and their differences- Different examples of control systems- Classification of control systems, Feed-Back Characteristics, Effects of feedback. Mathematical models – Differential equations, Impulse Response and transfer functions. **Transfer Function Representation:** Block diagram representation of systems considering electrical systems as examples -Block diagram algebra – Representation by Signal flow graph - Reduction using mason's gain formula.
2.	**Time Response Analysis:** Standard test signals - Time response of first order systems – Characteristic Equation of Feedback control systems, Transient response of second order systems - Time domain specifications – Steady state response - Steady state errors and error constants – Effects of proportional derivative, proportional integral systems.
3.	**Stability Analysis in S-Domain:** The concept of stability – Routh's stability criterion – qualitative stability and conditional stability – limitations of Routh's stability. **Root Locus Technique:** The root locus concept - construction of root loci-effects of adding poles and zeros to G(s) H(s) on the root loci.
4.	**Frequency Response Analysis:** Introduction, Frequency domain specifications-Bode diagrams-Determination of Frequency domain specifications and Phase margin and Gain margin-Stability Analysis from Bode Plots. Polar Plots, Nyquist Plots Stability Analysis. Compensation techniques – Lag, Lead, Lead-Lag Controllers design in frequency Domain, PID Controllers.
5.	**State Space Analysis of Continuous Systems:** Concepts of state, state variables and state model, derivation of state models from block diagrams, Diagonalization- Solving the Time invariant state Equations- State Transition Matrix and it's Properties – Concepts of Controllability and Observability.

Text Books

1. Control Systems Theory and Applications - S. K. Bhattacharya, Pearson.
2. Control Systems - N. C. Jagan, BS Publications.
3. Modern Control Systems by Ogatta
4. Automatic Control systems by B C Kuo

References

1. Control Systems - A. Ananad Kumar, PHI.
2. Control Systems Engineering - S. Palani, TMH.
3. Control Systems - Dhanesh N. Manik, Cengage Learning.
4. Control Systems Engineering - I. J. Nagrath and M. Gopal, New Age International (P) Limited, Publishers.
5. Control Systems - N. K. Sinha, New Age International (P) Limited Publishers.

Subject: Digital Electronics & Logic Design Lab	**Year & Semester: B. Tech** **Computer Science Engineering** **2nd Year & 4thSemester**	**Total Course Credit: 1**		
		L	T	P

(Code: ECL254)		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

To gain a firm understanding of concepts learned in the Digital electronics and Logic design course work by practical demonstration.

Learning Outcomes

The student should be able to construct digital circuits using standards ICs and testing boards.

Course Outline / Content	
Unit	**Topics**
1.	To verify the truth table of following logic gates: 1. AND OR and NOT 2. NAND, NOR, XOR and XNOR

2.	To design and realize Logic gates using universal gates.
3.	To design and realize:- a. Half adder and verify its truth table. b. Full adder and verify its truth table. c. Half subtractor and verify its truth table d. Full subtractor and verify its truth table.
4.	To design a multiplexer/demultiplexer using two input NAND/NOR gates
5.	Design and realize the following flip flops using logic gates. • RS flip flop • JK flip flop • D flip flop • T flip flop
6.	To design a modulo-10 counter.
7.	To design frequency dividing circuits.

Text Books

1. Digital Logic & Computer Design by M Morris Mano.
2. Digital Electronics by Gupta &Singhal.
3. Circuit Design with VHDL, V A Pedroni.

References

1. Digital principles and applications by A. P. Malvino
2. Switching Circuits by Marcus.

Subject: Communication System Lab (Code: ECL255)	**Year & Semester: B. Tech Computer Science Engineering 2nd Year & 4thSemester**	**Total Course Credit: 1**		
		L	T	P
		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Objectives

This course has experiments provides the foundation education in communication engineering lab analysis and design. Through lecture, laboratory, and out-of-class assignments, students are provided learning experiences that enable them to Analyze and deign basic electronic circuits, to carry out AM and FM modulation experiments using discrete electronic components and Become proficient with computer skills (eg., OrCADPspice and MATLAB) for the analysis and design of circuits .

Learning Outcomes

· Study signal and linear time invariant system properties.

· Study, design, and build amplitude modulation systems examining tradeoffs in different communication systems.

· Study, design, and build angle modulation systems examining tradeoffs in different communication systems.

· Perform experiments in converting analog information into digital data via sampling, quantization, and coding.

Course Outline / Content	
Unit	**Topics**
1.	**Use Network Analyzer for the following experiments**: · Measurement of transmission line parameters. · S-parameter estimation of Microwave devices. · Design and testing of a Microstrip coupler. · Characteristics of λ/4 and λ/2 transmission lines.

2.	**Use appropriate simulation tools for the following experiments**: · Channel equalizer design (LMS, RLS) · Antenna Radiation Pattern measurement. · Performance Evaluation of digital modulation schemes · OFDM transceiver design · Simulation of Microstrip Antennas · Performance evaluation of simulated CDMA System.
3.	· Exponential Fourier Series · Fourier Series using Matlab · Autocorrelation and Energy Spectral Density · Amplitude Modulation · Envelope Detection · Study the Basic Operation of Phase-Lock-Loop (PLL) · FM Modulation and Demodulation using PLL · Single Transistor FM Voice Transmitter · A Simple Sampler using 555 Timer · Pulse Width Modulation · Pulse Position Modulation

References

1. Communication Systems Laboratory Manual by Muhammad Tahir et al Department of Electrical Engineering University of Engineering and Technology Lahore.

Subject: Operating System Lab (Code: ITL252)	Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester	Total Course Credit: 1		
		L	T	P
		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Outcomes (COs):

To familiarize the students with:

CO1: Operating System commands.

CO2: Process Management and Control.

CO3: Implementation of Scheduling Algorithms.

CO4: Analysis of deadlocks in the Operating System.

CO5: System Programming and Kernel design.

List of Experiments:

1. To familiarize the students with the Operating Systems.
2. Introduction and use of basic Linux commands.
3. To demonstrate the process, memory, file and directory management modules under the Linux/Windows operating systems
4. To introduce Linux basic commands
5. To demonstrate use of Window APIs.
6. Write programs using the following system calls of UNIX operating system:

Fork, exec, getpid, exit, wait, close, stat, opendir, readdir

7. Write programs to implement Thread management using pthread library.
8. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time.

9. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time.
10. Write programs to simulate and analyze page replacement algorithms with respect to various parameters. Implement the Producer Consumer problem using semaphores.
11. Implement the deadlock free solution to Dining Philosophers problem to illustrate the problem of deadlock and/or starvation that can occur when many synchronized threads are competing for limited resources.
12. System Programming and Kernel design.

Subject: Database Management System Lab (Code: ITL253)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 1**		
		L	T	P
		0	0	2
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Outcomes (COs):

CO1: Transform an information model into a relational database schema and to use a data definition language and/or utilities to implement the schema using a DBMS.

CO2: Analyze the database using queries to retrieve records.

CO3: Formulate query, using SQL, solutions to a broad range of queries, and data update problems.

CO4: Applying PL/SQL for processing database.

CO5: Analyse front end tools to design forms, menus, etc and establish back-end connectivity

CO6: Develop solutions using database concepts for real-time requirements.

List Of Experiments:

1. Creation of a database and writing SQL queries to retrieve information from the database.

2. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.
3. Creation of Views for different users.
4. Creating an Employee database to set various constraints.
5. Creating relationship between the databases.
6. Study of PL/SQL block.
7. Creation of Procedures.
8. Creation of database triggers, cursors and functions.
9. Mini project (Application Development using Oracle/ Mysql/DB2)
    a) Inventory Control System.
    b) Material Requirement Processing.
    c) Hospital Management System.
    d) Railway Reservation System.
    e) Personal Information System.
    f) Web Based User Identification System.
    g) Timetable Management System.
    h) Hotel Management System

Subject: Web Programming (Code: ITL254)	**Year & Semester: B. Tech Information Technology Engineering 2nd Year & 3rd Semester**	**Total Course Credit: 2**		
		L	T	P
		0	0	4
Evaluation Policy	Mid-Term/Class Assessment (40 Marks)	Final-Term (60 Marks)		

Course Outcomes (COs):

CO1: Students will be able to create HTML Documents with formatting, images, tables, frames, embed multi-media objects and develop a static website using Hyper Text Mark-up Language.

CO2: Students will learn how to Create web pages using Cascading Style Sheets.

CO3: Students will design and implement dynamic websites with good aesthetic sense of designing and latest technical know-how's.

Syllabus:

UNIT I - HTML & Introduction to CSS

HTML for structure, CSS for layout, and JavaScript for client-side programming; Suggestions for learning. Web Site Basics: Dreamweaver, HTML: Elements. Attributes and values.HTML Tables: Table, heading, row, data elements and attributes. Table structure not for page layout.Links and server-side includes: HTML links and anchors. Linking to external files to modularize html, build script libraries, or share styles; Server-side Includes. Standards: W3C, the World Wide Web Consortium: W3C recommendations as standards.HTML rules: Extensible markup languages; Frames: A glance at a common but deprecated element; advantages and disadvantages; frame and frameset properties. Images:
Image types (JPG, GIF, PNG). Inline, embedded, and external styles. Writing Style Rules: Writing CSS selectors and rules to tie style attributes and values to html elements. The cascade: Inheritance, specificity, and the cascade.CSS positioning: Static, relative, and absolute positioning.

UNIT II - Introduction to JavaScript:

Client-side programming for browsers. Event Handlers. JavaScript Overview: Language characteristics. Variables. Assignment and comparison operators; expressions.HTML Forms: The form element and inputs: textbox, radio buttons, checkbox, textarea.

UNIT III - Advanced HTML & CSS:

HTML Form Basics, JavaScript, JavaScript Functions: Writing blocks of separate, reusable code, Getting started with developing simple functions for form validators. Form Validation: JavaScript for Simple Form Validation, The DOM and JavaScript Object Models: The W3C Document Object Model; using nodes; DHTML: JavaScript + CSS = Dynamic HTML, Advanced form validation: Javascript's innerHTML and dynamic CSS for advanced form validation

UNIT IV - JavaScript Programming:

Tracking the Mouse: Reporting the x and y position of the mouse, Annotating text: Adding hidden text and accessing through JavaScript, Advanced JavaScript—Super Hypertexts: Finding. JavaScript's Built-in Objects: Arrays. Dates. Math. Number and String Objects, Web Site Design / Redesign: Overview of site redesign. Client survey.

Text Books:

1. Dietel & Dietel "Internet & Web Designing".
2. John Duckett. "JavaScript and Jquery: Interactive Front-end Web Development".

Reference Books:

1. Greenlaw R and Hepp E "Fundamentals of Internet and www".
2. B. Underdahle and K.Underdahle, "Internet and Web Page / WebSite Design", IDG Books India (P) Ltd.
3. D. Comer, "The Internet Book", Prentice Hall of India.
4. David Flanagan. "JavaScript: The Definitive Guide".