# DATA COMMUNICATION

**ENCT 253**

| | | |
|---|---|---|
| **Lecture** | **: 3** | **Year :II** |
| **Tutorial** | **: 1** | **Part : II** |
| **Practical** | **: 3/2** | |

**Course Objectives:**

The objective of this course is to provide students with a solid foundation in the principles and theories of data communication, including key terminology, protocols, and standards. Also support to explore the various types of transmission media, including guided and unguided media, and their characteristics, advantages, and disadvantages. Furthermore, it introduces the methods of data encoding, modulation techniques, and their implications for effective data transmission.

**1 Introduction (4 hours)**

1.1 Analog data communication, data representation, data flows
1.2 Evolution of data communication
1.3 A communication model, data communication model
1.4 Networks (LAN, WAN), simplified network architecture, the OSI model
1.5 Data communication and networking for today enterprise

**2 Data Communication Fundamentals (6 hours)**

2.1 Analog and digital data
2.2 Analog signals, periodic and aperiodic signals, periodic signals characteristics (Time, frequency domain)
2.3 Introduction to Fourier series representation of periodic signal, Fourier transform representation of aperiodic signals, digital signals and its characteristics
2.4 Analog and digital transmission, transmission mode, transmission impairments (Attenuation, distortion, noise)
2.5 Data rate limits channel capacity, Nyquist bandwidth, Shannon capacity formula
2.6 Performance of network (Bandwidth, throughput, latency, jitter)

**3 Transmission Media and Data Compression (8 hours)**

3.1 Guided transmission media: Co-axial cable, twisted pair, optical fiber
3.2 Unguided transmission media: Radio waves, microwaves, infrared
3.3 Antenna basics, satellite communication, Bluetooth, Wi-Fi

3.4 Wireless propagation (Introduction to groundwave propagation, sky wave propagation and line of sight propagation), frequency bands
3.5 Error detection and correction: Parity, check sum, cyclic redundancy check, hamming code
3.6 Data compression: Lossy and lossless

**4 Signal Encoding Technique (15 hours)**

4.1 Analog data, analog signals: Modulation and its need, AM, FM, PM
4.2 Analog data, digital signals: PAM, PWM, PPM, PCM, DPCM, DM
4.3 Digital data, analog signal: ASK, FSK, PSK, QPSK, QAM
4.4 Digital data, digital signal: RZ, NRZ, AMI, Manchester, differential Manchester, B8ZS, HDB3 for data transmission

**5 Multiplexing and Switching (8 hours)**

5.1 Access introduction to multiplexing, application of multiplexing
5.2 Frequency division multiple
5.3 Time division multiple access
5.4 Asymmetric digital subscriber line, XDSL
5.5 Spread spectrum: DHSS, FHSS, CDMA
5.6 Intro switched communication network, connection oriented and connectionless
5.7 Switching devices: Types, importance and application
5.8 Circuit switching network: Circuit switching concepts, message switching
5.9 Packet switching: Virtual switching, datagram switching

**6 Cellular Wireless Communications and Latest Trends (4 hours)**

6.1 Overview of 1G, 2G, 3G and 4G
6.2 Cellular technology fundamental terminology: Cell, frequency-reuse, cluster, adjacent cell interference, co-channel interference, handoff strategies, architecture of GSM basics
6.3 Introduction to 5G networks, software defined networking, IOT communication, cloud computing and virtualization in data communication

**Tutorial (15 hours)**

1. Tutorials on different protocols in data communication TCP/IP, HTTP/HTTPS, FTP
2. Explore the function of open systems interconnection (OSI) model, which defines seven layers of data communication
3. Discover data communication devices and its application
4. Identify the application of used network topologies in present scenario
5. Collecting ideas on some security aspects of the security on data communication on present enterprises system

प्राविधिक परिषद्को कार्यालय
कीर्तिपुर

Institute of Engineering
Office of the Dean, Pulchowk

**Practical** **(22.5 hours)**

1. Signal analysis using MATLAB
2. Implementation of small network using hub and switch in physical or simulation environment
3. Analog modulation generation and reconstruction
4. Pulse modulation generation and reconstruction
5. Conversion of given binary sequence into different line coding
6. Digital modulation (ASK, FSK, PSK) generation and reconstruction

**Final Exam**

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

| Chapter | Hours | Mark distribution* |
|---|---|---|
| 1 | 4 | 5 |
| 2 | 6 | 9 |
| 3 | 8 | 11 |
| 4 | 15 | 20 |
| 5 | 8 | 10 |
| 6 | 4 | 5 |
| **Total** | **45** | **60** |

* There may be minor deviation in marks distribution.

**Reference**


1. Stallings, W. (2007). Data and Computer Communications. India: Pearson Education.
2. Forouzan, B. A., Fegan, S. C. (2007). Data Communications and Networking (McGraw-Hill Forouzan Networking). United Kingdom: McGraw-Hill Higher Education.
3. Tanenbaum, A. S., Wetherall, D. (2011). Computer Networks. India: Pearson Prentice Hall.
4. Rappaport, T. S. (2024). Wireless Communications: Principles and Practice. United Kingdom: Cambridge University Press.

# DATA STRUCTURE AND ALGORITHMS

**ENCT 252**

**Lecture : 3** **Year : II**
**Tutorial : 1** **Part : II**
**Practical : 3**

**Course Objectives:**
The objective of this course is to impart fundamental knowledge on the design and implementation of data structures for storing information. It also covers various algorithms used in computer science. Upon completion of this course, students will be able to design and choose the appropriate data structure and efficient algorithm to achieve optimal performance

**1 Introduction** **(4 hours)**

1.1 Introduction to data structures
  1.1.1 Need of data structures
  1.1.2 Types of data structures and its characteristics
1.2 Abstract data type (ADT)
1.3 Basics of algorithm design techniques (Brute Force, divide and conquer, Greedy algorithms, branch and bound, backtracking, randomized, recursive, dynamic programming)
1.4 Algorithm analysis
  1.4.1 Time and space complexity
  1.4.2 Best, worst and average case analysis
  1.4.3 Rate of growth
  1.4.4 Asymptotic notations: Big Oh, Big Omega and Big Theta

**2 Stack and Recursion** **(7 hours)**

2.1 Definition of stack and its operations
2.2 Array implementation of stack ADT
2.3 Stack applications
  2.3.1 Expression conversion: Infix to postfix and prefix expression
  2.3.2 Expression evaluation: Infix and postfix expression evaluation
2.4 Recursion
  2.4.1 Concept of recursion
  2.4.2 Recursion and stack
  2.4.3 Recursion vs iteration
  2.4.4 Execution of recursive calls
  2.4.5 Types of recursions
  2.4.6 Applications of recursion: Tower of Hanoi

## 3 Queues (5 hours)

3.1 Definition of queue and its operations
3.2 Array implementation of queue ADT
3.3 Types of queue ADT: Linear, circular, double ended and priority queues

## 4 Linked List (6 hours)

4.1 Definition of list and its operations
4.2 Array implementation of list ADT
4.3 Static list and its limitations
4.4 Linked list: Definition and its operations
4.5 Types of linked list: Singly, doubly, circular
4.6 Application of linked list
  4.6.1 Linked list implementation of stack and queue ADT
  4.6.2 Solving polynomial equations using linked list

## 5 Tree (7 hours)

5.1 Definition and tree terminologies
5.2 Binary trees
  5.2.1 Definition and types
  5.2.2 Array and linked list representation
  5.2.3 Traversal algorithms: Pre-order, in-order and post-order traversal
  5.2.4 Application of full binary tree: Huffman algorithm
5.3 Binary search tree
  5.3.1 Definition and operations on binary search tree: Insertion, deletion, searching and traversing
  5.3.2 Construction of binary search tree
5.4 Balanced binary tree
  5.4.1 Problem with unbalanced binary trees
  5.4.2 Balanced binary search tree
  5.4.3 AVL tree, definition and need of AVL tree, construction of AVL tree: Insertion, deletion on AVL tree and rotation operations
5.5 Introduction to red-black tree
5.6 B-Tree: Need, definition and construction of B-tree

## 6 Graphs (6 hours)

6.1 Definition, terminologies and types of graphs
6.2 Representation of graphs: Adjacency matrix, incidence matrix and adjacency list
6.3 Transitive closure and Warshall's algorithm
6.4 Graph traversals: Breadth-first search, depth-first search and topological sort

6.5 Minimum spanning tree: Kruskal's algorithm and prim's algorithm
6.6 Shortest-paths problems: Dijkstra's algorithm, Floyd- Warshall algorithm

**7 Sorting Algorithms (5 hours)**

7.1 Definition of sorting and its applications
7.2 Types of sorting: Internal/external sort, stable/unstable sort, in-place/ not in-place sort, adaptive/ non-adaptive sort
7.3 Sorting algorithms and its efficiency: Bubble, insertion, selection, shell, quick, merge, radix and heap sorting

**8 Searching Algorithms (5 hours)**

8.1 Definition of searching techniques and its applications
8.2 Different searching algorithms and its efficiency
  8.2.1 Sequential search
  8.2.2 Binary search
8.3 Hashing
  8.3.1 Definition and its applications
  8.3.2 Hash function
  8.3.3 Hash table
  8.3.4 Collision in hash table
  8.3.5 Collision resolution techniques: Chaining method and open addressing method (Linear probing, quadratic probing and double hashing)

**Tutorial (15 hours)**

1. Analyzing time and space complexity of basic algorithms and comparing best, worst, and average cases with examples
2. Solving problems using stacks: Converting infix expressions to postfix/prefix, evaluating postfix expressions, balancing parentheses, reversing a string
3. Solving Tower of Hanoi recursively and analyzing tail vs non-tail recursion
4. Solving polynomial addition using linked lists
5. Implementing tree traversals: Preorder, inorder, postorder
6. Constructing a binary search tree (BST) and performing insertion, deletion, and searching, balancing BSTs with AVL tree operations (Rotations, insertion, deletion), constructing B-Trees (Insertion, deletion)
7. Implementing Huffman coding for text compression
8. Solving shortest path problems using Dijkstra's and Floyd-Warshall algorithms
9. Implementing Kruskal's and Prim's algorithms for minimum spanning tree
10. Analyzing and comparing sorting algorithm efficiencies
11. Implementing linear search and binary search
12. Designing a hash table with different collision resolution techniques (Chaining, linear probing, quadratic probing, double hashing)

**Practical** **(45 hours)**

1. Implementation of stack using array, applications of stack- conversion of infix to prefix and postfix expression, evaluation of prefix and postfix expression, matching parenthesis, reversal of string
2. Implementation of recursive algorithms (Tail and non-tail method)- Factorial, sum of natural numbers, Fibonacci series, implementation of Tower of Hanoi
3. Implementations of linear queue and circular queue using arrays
4. Implementation of static list, implementation of linked list: Singly and doubly linked lists
5. Implementation of stack and queue using linked list, application of linked list- polynomial addition
6. Implementation of in-order, pre-order and post-order tree traversals
7. Implementation of breadth-first and depth-first search to traverse a graph
8. Implementation of different sorting algorithms
9. Implementation of different searching algorithms

**Final Exam**

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

| Chapter | Hours | Marks distribution* |
|---|---|---|
| 1 | 4 | 5 |
| 2 | 7 | 9 |
| 3 | 5 | 7 |
| 4 | 6 | 8 |
| 5 | 7 | 9 |
| 6 | 6 | 8 |
| 7 | 5 | 7 |
| 8 | 5 | 7 |
| **Total** | **45** | **60** |

* There may be minor deviation in marks distribution.

**Reference**


1. Langsam, Y. Augenstein .M. J. and Tenenbaum A. M. (1996). Data Structures using C and C++. Prentice Hall Press.
2. Rowe, G. W. (1997). Introduction to data structures and algorithms with C++. Prentice-Hall, India.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C, (2022). Introduction to algorithms. MIT press.
4. Kruse, R. L., and Ryba, A. J., (1998). Data structures and program design in C++. Prentice Hall, India.
5. Thareja, R. (2014). Data Structures Using C. Oxford University Press.

# DISCRETE STRUCTURE

**ENCT 251**

| | | |
|---|---|---|
| **Lecture** | **: 3** | **Year : II** |
| **Tutorial** | **: 1** | **Part : II** |
| **Practical** | **: 0** | |

**Course Objectives:**

The objective of this course is to provide basic understanding in discrete mathematics and finite state machine. It also emphasizes to build fundamental and conceptual clarity in the area of logic, reasoning, proof, recurrence relation, graph theory, theory of automata and algorithmic analysis.

**1 Logic and Induction (8 hours)**

1.1 Review of set theory, relation and function

1.2 Proposition, connectives in proposition, types of propositions, truth function and propositional logic

1.3 Expressing statements in logic propositional logic, rules of inference in propositional logic, validity of an argument, methods of tableaux

1.4 Predicate logic and quantification, informal deduction in predicate logic

**2 Proof Techniques (5 hours)**

2.1 Formal proofs and informal proofs, mathematical reasoning- direct proof and indirect proof (Proof by contradiction and proof by contraposition)

2.2 Elementary induction and complete induction, strong induction

2.3 Proof by counter example, vacuous and trivial proofs, proof by cases, mistakes in proof

**3 Automata Theory, Regular Language and Grammar (10 hours)**

3.1 Alphabet, string, string operations and language, introduction to finite automata

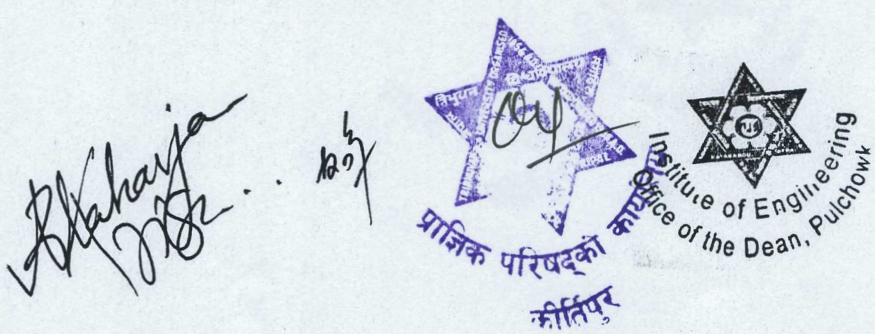
3.2 Deterministic finite automata (DFA), representation and language of DFA

3.3 Non deterministic finite automata (NFA), equivalence of DFA and NFA

3.4 Regular expressions and its characteristics, regular language and its properties

3.5 Equivalence of regular expression and finite automata

3.6 Context free grammar and context free language



## 4 Recurrence Relation and Algorithmic Analysis (7 hours)

4.1 Recurrence relations, recurrence relation for tower of Hanoi (TOH) and Fibonacci series, solving linear recurrence relations (Homogeneous and non-homogeneous)

4.2 Algorithm and its properties, asymptotic notation of algorithm

4.3 Linear and binary search and their analysis; Bubble and insertion sorting and their analysis

## 5 Graph Theory and Tree (15 hours)

5.1 Graphs basics, graph terminologies, graph types (Directed, un-directed, simple, weighted, regular, complete, bipartite, planar graph) and special graphs

5.2 Subgraphs, graph representation, connectivity in graphs and its components, strongly and weakly connected graphs

5.3 Paths and circuits, Euler path and circuit, Hamiltonian path and circuit

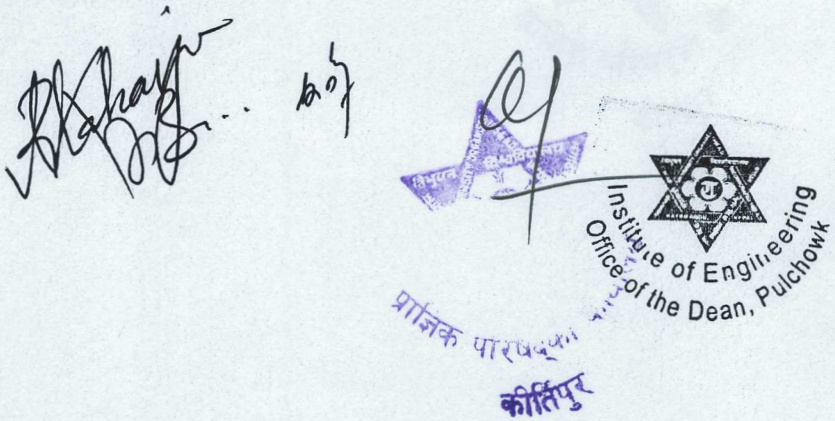
5.4 Shortest path algorithm (Dijkstra's algorithm), graph coloring and four color theorem, applications of graph coloring.

5.5 Graph as network, maximal flows and minimal cuts, the max flow-min cut theorem

5.6 Introduction and applications, tree traversals, spanning trees, minimum spanning trees (Prim's and Kruskal's algorithm)

## Tutorial (15 hours)

1. Problem related to function, relation and validity of an arguments using rules of inference in propositional and predicate logic
2. Problem related to proof techniques
3. Design of automata, conversion from NFA to DFA, defining grammar for language and properties of regular language
4. Problem related to different recurrence relation including TOH, Fibonacci series and homogeneous and non-homogeneous recurrence relation
5. Representation of graph, finding shortest path, obtaining minimum spanning tree, problems related to planar graph, graph coloring and maximum flow
6. Defining the complexity of liner and binary search, problems related to bubble and insertion sorting algorithms



**Final Exam**

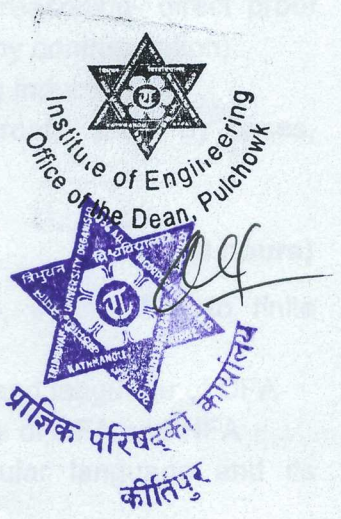
The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

| Chapters | Hours | Marks distribution* |
|---|---|---|
| 1 | 8 | 10 |
| 2 | 5 | 7 |
| 3 | 10 | 14 |
| 4 | 7 | 9 |
| 5 | 15 | 20 |
| **Total** | **45** | **60** |

* There may be minor deviation in marks distribution.

**References**


1. Rosen, K. H. (2019). Discrete Mathematics and Its Applications. United Kingdom: McGraw-Hill.
2. Johnsonbaugh, R. (2007). "Discrete Mathematics", Prentice Hall Inc.
3. Chartrand, G., Oellermann, O. R. (1993). Applied and Algorithmic Graph Theory. Singapore: McGraw-Hill.
4. Lewis, H. R., Papadimitriou, C. H. (1981). Elements of the Theory of Computation. United Kingdom: Prentice-Hall.
5. Cormen, T. H., Leiserson, C. E., Stein, C., Rivest, R. L. (2009). Introduction to Algorithms, Third Edition. France: MIT Press.

# OPERATING SYSTEM

**ENCT 254**

| | | |
|---|---|---|
| **Lecture** | **: 3** | **Year : II** |
| **Tutorial** | **: 1** | **Part : II** |
| **Practical** | **: 3/2** | |

**Course Objectives:**

The objective of this course is to familiarize students with the different aspects of operating systems and encourage them to use these ideas in designing operating systems.

## 1 Introduction (6 hours)

- 1.1 Introduction to operating systems
- 1.2 OS as an extended machine and resource manager
- 1.3 History of operating system
- 1.4 Type of operating system: Mainframe, server, personal, smartphone and handheld, IOT and embedded, real-time, smart-card
- 1.5 Operating system components: Kernel, shell, utilities, applications
- 1.6 Types of OS kernel: Monolithic, micro, nano, layered, hybrid, exo-kernel
- 1.7 System calls, shell commands, shell programming
- 1.8 POSIX standard
- 1.9 Bootloader, MBR/GPT, UEFI and legacy boot

## 2 Process Management (7 hours)

- 2.1 Process description, states and control
- 2.2 Scheduling algorithms
  - 2.2.1 First Come First Serve (FCFS)
  - 2.2.2 Shortest Job First (SJF)
  - 2.2.3 Shortest Remaining Time(SRT)
  - 2.2.4 Round Robin (RR)
  - 2.2.5 Highest Response Ratio Next (HRNN)
  - 2.2.6 Completely Fair Scheduler (CFS) used in Linux
- 2.3 Threads and thread scheduling

## 3 Process Communication and Synchronization (10 hours)

- 3.1 Principles of concurrency, race condition, critical region
- 3.2 Mutual exclusion, semaphores, and mutex
- 3.3 Message passing and monitors
- 3.4 Classical problems of synchronization: Readers-writers problem, producer-consumer problem, dining philosopher problem
- 3.5 Deadlock: Prevention, ignorance, avoidance, detection and recovery

**4 I/O and Memory Management (9 hours)**

4.1 I/O management
- 4.1.1 Principles of I/O hardware and software
- 4.1.2 I/O software layer
- 4.1.3 Disk technologies: Magnetic disk, SSD, NVMe storage
- 4.1.4 RAID
- 4.1.5 Concept of stable storage, cost per bit comparison

4.2 Memory Management
- 4.2.1 Memory address, swapping and managing free memory space
- 4.2.2 Virtual memory management, paging, segmentation
- 4.2.3 Page replacement algorithms (FIFO, LRU, LFU), page fault and hit ratio
- 4.2.4 Allocation of frames
- 4.2.5 Thrashing

**5 File Systems (3 hours)**

5.1 File concepts: Name, structure, types, access, attributes, operations
5.2 Directory structures: Paths and hierarchies (Linux/Windows)
5.3 File system implementation: Inodes, allocation methods (Contiguous, linked, indexed)
5.4 File system performance: Factors affecting efficiency
5.5 Example file systems: NTFS, EXT4, FAT32, NFS

**6 Security and System Administration (3 hours)**

6.1 OS security: Cryptography, multi-factor authentication (MFA), secure boot and sandboxing
6.2 Access control: Policies, lists, and OS support
6.3 System administration: User management, environment setup and tools (AWK, shell scripts, make)

**7 Hypervisors and Virtual Systems (4 hours)**

7.1 Hypervisors: Type 1 and type 2
7.2 Virtual machines: Creating virtual machine in Qemu/Virtual box/VMWare
7.3 Container virtualization: Docker and Kubernetes
7.4 Power shell and windows subsystem for LINUX (WSL)
7.5 Performance optimization and security in virtualized environments

**8 Overview of Contemporary OS (3 hours)**

8.1 Windows and Linux-based OS
8.2 Embedded and mobile OS
8.3 IoT and RT operating system
8.4 Robot and smart card operating system

**Tutorial** **(15 hours)**

1. Unix shell programs to do a variety of tasks
2. Problems on scheduling algorithms
3. Problems on page replacement algorithms
4. Problems on deadlock
5. Problems on segmentation and paging
6. Comprehensive study on storage technologies with cost/bit and speed comparisons
7. Study on process management/memory management/IO management/ security/file system of a selected OS

**Practical** **(22.5 hours)**

1. Basic Unix commands
2. Shell Programming
3. Implementation of the ls and grep commands
4. Programs using the I/O system calls of the UNIX operating system
5. Implementation of scheduling algorithms and the producer-consumer problem using semaphores
6. Implementation of some memory management schemes such as paging and segmentation
7. Term Project

**Final Exam**

The questions will cover all the chapters in the syllabus. The evaluation scheme will be as indicated in the table below:

| Chapter | Hours | Marks Distribution* |
|---|---|---|
| 1 | 6 | 8 |
| 2 | 7 | 10 |
| 3 | 10 | 13 |
| 4 | 9 | 12 |
| 5 | 3 | 4 |
| 6 | 3 | 4 |
| 7 | 4 | 5 |
| 8 | 3 | 4 |
| **Total** | **45** | **60** |

* There may be minor deviation in marks distribution.

**Reference**


1. Tanenbaum, A.S., Bos, H. (2024). Modern Operating Systems .3rd Edition. PHI.
2. Stalling, W. (2008). Operating Systems. 6th Edition. Pearson Education.
3. Chaturvedi, A., Rai, B.L. (2017). UNIX and Shell Programming. University Science Press.