5th Semester

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course Title	Code	L	T	P	Credits
Design & Analysis of Algorithms	**ITT301**	**3**	**1**	**0**	**4**

Course Outcomes (COs):

CO1: Understand basics of algorithm efficiency and asymptotic notations.
CO2: Study various divide & conquer and greedy algorithms.
CO3: Understand the concept of dynamic programming with applications.
CO4: Study various graph searching and traversal algorithms.
CO6: Understand various computational complexity measures.

Syllabus:

UNIT I - Introduction

Algorithm Design paradigms - motivation, concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations, Master theorem.

UNIT II - Divide & Conquer methods

Divide & Conquer algorithms: examples, Binary search, Quick sort, Strassen's algorithm for matrix multiplication, analysis of divide and conquer runtime reference relations.

UNIT III - Dynamic Programming and Greedy paradigm

Overview of dynamic programming, difference between dynamic programming and divide and conquer. Dynamic Programming: Matrix Chain Multiplication (MCM), Longest Common Subsequence (LCS), Optimal Binary Search Tree (OBST). Overview of the greedy paradigm. General Greedy approach Vs Dynamic Programming approach Case studies: fractional Knapsack vs 0/1 Knapsack problem.

UNIT IV - Graph searching and traversal

Representation of Graphs, Breadth First Search, Depth First Search, Topological Sort, Strongly Connected Components, examples of exact optimization solution (minimum cost spanning tree), Dijkstra's and Bellman ford algorithm, All pair shortest path, Flyod Warshall Algorithm.

UNIT V - Backtracking and Computational complexity

Back Tracking: Overview, 8-queen problem. Branch & Bound: LC searching, bounding, FIFO branch and bound, Travelling salesman problem. Computational complexity Complexity

measures, Polynomial vs non-polynomial time complexity; NP hard and NP complete classes, examples

Text Books:

1. Introduction to Algorithms, by Cormen, Leiserson, Rivest, and Stein, MIT Press.

Reference Books:

1. Algorithms, by Dasgupta, Papadimitrou and Vazirani, McGraw-Hill Education, 2006.
2. Computer Algorithms, by Horowitz, Sahni, and Rajasekaran, Silicon Press, 2007.
3. Algorithm Design, by Kleinberg and Tardos, Pearson, 2005.
4. Algorithm Design, by Goodrich and Tamassia, Wiley, 2001.

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course	Code	L	T	P	Credits
Microprocessor	**ITT302**	**3**	**1**	**0**	**4**

Course Outcomes (COs):

CO1: Describe the general architecture & organization of 8085 and 8086 Microprocessor and understand the difference between 8085 and advanced microprocessor.
CO2: Understand and classify the instruction set of 8085 and 8086 microprocessors and distinguish the use of different instructions and apply it in assembly language programming.
CO3: Ability to understand and write programs for stacks, delays, counters and subroutines.
CO4: Illustrate how the different peripherals (8255, 8279, etc.) are interfaced with Microprocessor
CO5: Analyze the data transfer information through serial and parallel ports.

Syllabus:

UNIT I - MICROPROCESSOR-BASED SYSTEMS: HARDWARE AND INTERFACING:

Microprocessors, Microcomputers, and Assembly Language, Introduction to 8085 Assembly Language Programming, Microprocessor Architecture and Microcomputer Systems, 8085 Microprocessor Architecture and Memory Interfacing Interfacing I/O Devices

UNIT II - PROGRAMMING THE 8085:

Introduction to 8085 Instructions, Programming Techniques with Additional Instructions, Counters and Time Delays, Stack and Subroutines, Code Conversion, BCD Arithmetic, and 16-Bit Data Operations, Software Development, Assemblers, and IDE

UNIT III - INTERFACING PERIPHERALS (I/OS) AND APPLICATIONS:

Interrupts, Interfacing Data Converters, Programmable Interface Devices: 8155 I/O and Timers: 8279 Keyboard / Display Interface, General Purpose Programmable Peripheral Devices, Serial I/O and Data Communication, Microprocessor Applications, Trends in Microprocessor Technology

UNIT IV - MICROPROCESSOR 8086:
Pin diagram, Architecture, Addressing Modes, Timing diagram, Instruction Set, Programming Techniques, Interrupt, Assembler Directives, Memory & I/O mapping

Text Books:

1. Ramesh S.Goankar, Microprocessor Architecture, Programming and Applications with the 8085.

Reference Books:

1. Douglas .V Hall, Microprocessor & Interfacing, Tata McGraw Hill
2. Rafiquzzuman .M, Microprocessor theory & Applications, Prentice Hall of India

3. Yuchenhiu, Glenn A Gibson, Microprocessor Systems - 8086/8088 Family, Prentice Hall of India

Course	Code	L	T	P	Credits
Computer Organisation & Architecture	**IT T303**	**3**	**1**	**0**	**4**

Course Outcomes (COs):

CO1: Understand the basics of computer architecture and how it interacts with the software. Understand how computers represent and manipulate data. Understand computer arithmetic.

CO2: Understand how decisions made in hardware affect the software/programmer as well as hardware designer.

CO3: Understand the fundamental principles and tradeoffs in designing the hardware/software interface i.e., instruction set architecture.

CO4: Understand the design of major components of a modern programmable microprocessor.

CO5: Understand the techniques to improve the performance of the modern processors.

CO6: Understand the basics of the memory hierarchy in the high performance computers and the numerous techniques to improve the efficiency of the memory system.

Syllabus:

UNIT I - INTRODUCTION TO COMPUTER ARCHITECTURE AND ORGANIZATION:
Defining computer architecture and computer organization, classes of computers, basic structure of computers, Operational concepts, performance and Amdhal's law.

UNITY II - ARITHMATIC AND LOGIC UNIT:
Microperations and their RTL specifications, Adder/Subtractor, Shifter, Multiplication and division circuits, Arithmatic logic shift unit.
Arithmetic addition & Subtraction of Signed and unsigned numbers-algorithm and hardware, Multiplication and division of Signed and unsigned numbers-algorithm and hardware, IEEE754 representation of Floating Point Numbers & Operations.

UNIT III - CONTROL AND PROCESSOR UNT:
Control Unit: Machine instructions, Execution of a complete Instruction, Multiple Bus organization, Hardwired control, Micro-programmed control.
Processor Unit: Components, organization types, addressing modes, Instruction types, Concept of sub-routine and sub-routine call. Use of stack.

UNIT IV - I/O AND MEMORY UNIT:
I/O Unit: Synchronous vs. Asynchronous I/O, I/O techniques - interrupts, polling, DMA, IOP
Memory unit: Memory organization, Types of memories and performance considerations, organization of memory modules, associative memory, cache memory and related mapping and replacement policies, virtual memory.

Introduction to Pipelining: Concepts, Basic pipelining, Hazards.

Text Books:

1. Computer Organization, Hamachar, Vranesic & Zaky, TMH.
2. Computer Organization & Architecture, M. M. Mano, PHI.

Reference Books:

1. Computer system architecture, Morris Mano, Pearson.
2. Computer organisation & Architecture, Paterson.

Course	Code	L	T	P	Credits
Theory of Computation	**ITT304**	**3**	**1**	**0**	**4**

Course Outcomes (COs):

CO1: Explore the different ways to reason about the correctness of algorithms for solving various computer science problems?

CO2: Defining the working and properties of various computational models. How do we mathematically model computers?

CO3: Designing finite automata and regular expressions, writing context-free grammars, reducing problems to one another.

CO4: Explore why some problems are harder to solve than others, and see how to reason with mathematical certainty.

CO5: Find the limits of what problems can be solved by computers. Proving which problems are impossible to solve with computers. Exploring $P \stackrel{?}{=} NP$.

Syllabus:

UNIT I - INTRODUCTION:

Mathematical Preliminaries and Notation, Sets, Relations, and Functions, Graphs, Methods of Proof, Basic Concepts: Languages, Grammars, Automata, some applications. Finite State Automata: Deterministic Finite acceptors, Deterministic acceptors and transitions Graphs, Languages and DFAs, regular languages, Non-deterministic Finite Acceptors, Definition, why Nondeterminism, Equivalence of NFA and DFA, Reduction of finite automata.mealy and Moore machines,

UNIT II - REGULAR LANGUAGES AND REGULAR GRAMMARS:

Regular expressions, definition, language associated with regular expressions, connection between Regular expression and regular languages, regular grammars, right and left linear grammars. Closure properties of regular languages under various operations, identifyingNonregular languages. Pigeonhole principle, pumping lemma.

UNIT III - CONTEXT-FREE LANGUAGES:

Definition of context free grammars, examples, leftmost and rightmost derivations, derivation Tree, Parsing and ambiguity, parsing and membership, ambiguity in grammars and languages. Context Free languages and programming languages. Methods for transforming grammars, substitution rules, removing useless productions, removing λ-productions, Removing Unit productions, Normal forms, Chomsky form, Greibach Normal form, Membership algorithms for context fee grammars. Properties of CFL, pumping lemmas, closure properties and decision algorithm properties, decidable properties of CFL.

UNIT IV - PUSHDOWN AUTOMATA:

Definition of Pushdown Automata, Nondeterministic Pushdown Automata, languages accepted by PDA, PDA for CFL,CFL for PDA, DPDA and DCFL. Grammars for CFLs.

Turing Machines: The Standard Turing Machine, Definition of a TuringMachine, Turing Machines as Language AcceptersTuring Machines as Transducers, Combining Turing Machines for Complicated Tasks, Turing's Thesis, variations on Turing machine. Nondeterministic Turing machine

UNIT V - UNDECIDABILITY:

The ChomskyHierarchy, Recursive and Recursively Enumerable Languages, Context-Sensitive (grammars and Languages, A language that is not Recursively Enumerable (RE),problems that cannot be solved by using Turing machine, An undecidable problem that isRE, Undecidable problems about Turing Machine, Post's Correspondence Problem ,the complexity classes P and NP and language families.

Text Books:

1. Peter Linz, "An Introduction to Formal Language and Automata", Narosa Publishing house.
2. M.Sipser; Introduction to the Theory of Computation; Singapore: Brooks/Cole, Thomson Learning.
3. John.C.martin, "Introduction to the Languages and the Theory of Computation", Tata McGrawHill.
4. K.Krithivasan and R.Rama; Introduction to Formal Languages, Automata Theory and Computation; Pearson Education.
5. J.E.Hopcroft, R.Motwani and J.D.Ullman , "Introduction to Automata Theory Languages and computation", Pearson Education Asia.

Course	**Code**	**L**	**T**	**P**	**Credits**

Data Communication **ITT305** **3 1 0 4**

Course Outcomes (COs):

CO1: Understand the basics of data and signal.
CO2: Study OSI and TCP/IP reference models and compare the two.
CO3: Discuss the different types of network topologies and types of networks based on size with suitable applications.
CO4: Explore the existing types of transmission media and compare them with the state of the art.
CO5: Study various techniques of analog and digital conversions.
CO6: Understand various techniques used in the physical layer and data link layer.

Syllabus:

UNIT I - DATA COMMUNICATION NETWORK:
Data communication concept, Basic concept of network, Types of networks (LAN, MAN and WAN), Different network topologies like star, ring, hybrid, tree. Network models (OSI and TCP/IP).
Transmission media: Guided and unguided media, twisted wire pair, co-axial cable, optical fibre, microwave links, satellite microwave link, their characteristic features and applications for data transmission.
Data and signals: Data, Signals, Types of Signals, Bandwidth, spectrum, transmission impairments, Shanon capacity.

UNIT II - DIGITAL TRANSMISSION TECHNIQUES:
Digital-to digital conversions: NRZ, RZ, Biphase, Manchester coding, AMI. Analog-to-digital conversions: Nyquist sampling theorem, quantization, Pulse code modulation, Delta modulation.

UNIT III - ANALOG TRANSMISSION TECHNIQUES:
Digital-to-analog conversion: ASK, FSK, PSK, QAM. Signal constellation. Analog-to-analog conversion: amplitude modulation, frequency modulation, phase modulation.

UNIT IV - BANDWIDTH UTILIZATION TECHNIQUES:
Frequency Division Multiplexing, Time Division Multiplexing, Wavelength division Multiplexing, Spread Spectrum.

UNIT V - ERROR DETECTION AND CORRECTION:
Errors in data communication: Types of errors, error detection and correction techniques, simple

parity check, computation of CRC, Checksum, Hamming code.

Recommended Books:

1. William Stallings: Data & Computer Communications, PHI.
2. Andrew Tanenbaum, "Computer Networks" PHI
3. Sklar, "Digital Communications fundamentals & Applications".
4. Keizer, " Local Area Networks" McGraw Hill

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course	Code	L	T	P	Credits
Introduction to Probability and Statistics	**MAT301**	**3**	**0**	**0**	**3**

Course Outcomes (COs):

CO1: Understand the basic concepts of random variables, probability distribution.
CO2: Understand concepts behind different distributions and their applications.
CO3: Understand the concept of joint probability distribution, Correlation Coefficient, Transformation of random variables,Regression Analysis
CO4: Compute point estimation of parameters, explain sampling distributions, and understand the central limit theorem.
CO5: Construct confidence intervals on parameters for a single sample.

Syllabus:

Unit-I Random variables:

Discrete and Continuous Random variables, Distribution functions, Expectation and Variance of Probability distribution, and Moment Generating function, Moments and properties. Discrete distributions: Binomial, Poisson and Geometric distributions and their applications.

Continuous distribution: Uniform, Exponential and Normal distributions, Normal approximation to Binomial distribution and their applications.

Unit II: Two-Dimensional Random Variables

Bivariate Random Variables, Joint Distribution Functions (Discrete and Continuous), Marginal and Conditional Distributions, Covariance and Correlation Coefficient, Transformation of random variables. Regression Analysis, Linear and Non linear Regression, Multiple regression, Curve fitting by method of least squares, fitting of straight lines, polynomials, exponential curves.

Unit III: Sampling Theory

Population and Sample, Statistical inference, Sampling with and without replacement, Random samples, Population parameters, Sample statics, Sampling distributions, Sample mean, Sampling distribution of means, Sample variances, Sampling distribution of variances, Case where population variances is unknown, Unbiased estimates and efficient estimates, point estimate and Interval Estimates, Confidence Interval estimates of population parameters.

Textbooks Recommnded:

1. Introductory STATISTICS, Neil A. Weiss, 9th Edition. Pearson, 2012.
2. Probability and Statistics for Engineers, Johnson, Miller and Freund, Pearson Education, 8th Edition, 2015.
3. Fundamentals of Statistics, S. C. Gupta, 7th Edition, Himalaya Publishing House 2018.
4. S. Ross: A First Course in Probability, 6th Ed., Pearson Education India, 2002.
5. Fundamentals of Mathematical Statistics, S.C. Gupta, V.K Kapoor, Sultan Chand,
6. Introduction to Mathematical Statistics, Robert V. Hogg, Joseph W. McKean and Allen T. Craig, Second Edition, LPE Pearson Prentice hall, 2007.

References:

1. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Third Edition, Narosa Pub. House, 2008.
2. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000
3. An Introduction to Probability and Mathematical Statistics, V.K. Rohatgi and A. K. Md. Ehsanes Saleh, Second Edition, John Wiley and sons, 2008.
4. Schaum's Outline ofTheory and Problems of Probability, Random Variables, and Random Processes, Hwei P. Hsu, Tata Mc-Graw Hill Edition.

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course Title	Code	L	T	P	Credits
Design & Analysis of Algorithms Lab	**ITL306**	**0**	**0**	**2**	**1**

Course Outcomes (COs):

CO1: Implement various divide and conquer based algorithms.
CO2: Study and implement greedy algorithms for minimum cost spanning tree, Knapsack problem, single source shortest paths.
CO3: Implement dynamic programming.
CO4: Implement various graph searching and traversal algorithms.
CO6: Implement branch and bound algorithm for various problems.

Syllabus:

1.	Divide and conquer algorithms.
2.	Greedy algorithms for minimum cost spanning tree, Knapsack problem, single source shortest paths.
3.	Dynamic Programming with applications.
4.	Graph searching and traversal algorithms.
5.	Backtracking algorithms: 8-queen problem and Knapsack problem.
6.	Branch and bound algorithm with applications.

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course	Code	L	T	P	Credits
Microprocessor Lab	**ITL307**	**0**	**0**	**2**	**1**

Course Outcomes (COs):

CO1: To become familiar with the architecture and Instruction set of Intel 8085 microprocessor
CO2: To provide practical hands on experience with Assembly Language Programming.
CO3: Develop ALP for 8 and 16 bit Arithmetic operations using 8086 microprocessor.
CO4: To familiarize the students with interfacing of various peripheral devices with 8085 microprocessor.
CO5: Analyze the data transfer information through serial & parallel ports.
CO6: To improve programming logic and concepts of 8085 microprocessor by developing programs for various applications.

Syllabus:

i) To develop a program to add two double byte numbers.
ii) To develop a subroutine to add two floating point quantities.
iii) To develop program to multiply two single byte unsigned numbers, giving a 16 bit product.
iv) To develop subroutine which will multiply two positive floating points numbers?
v) To write program to evaluate P* Q*+R* & S are 8 bit binary numbers.
vi) To write a program to divide a 4 byte number by another 4 byte number.
vii) To write a program to divide an 8 bit number by another 8 bit number upto a fractional quotient of 16 bit.
viii) Write a program for adding first N natural numbers and store the results in memory location X.
ix) Write a program which decrements a hex number stored in register C. The Program should half when the program register reads zero.
x) Write a program to introduce a time delay of 100 ms using this program as subroutine display numbers from 01H to OAH with the above calculated time delay between every two numbers.
xi) N hex numbers are stored at consecutive memory locations starting from X. Find the largest number and store it at location Y.

xii) Interface a display circuit with the microprocessor either directly with the bus or by using I/O ports. Write a programme by which the data stored in a RAM table is displayed.

xiii) To design and interface a circuit to read data from an A/D converter, using the 8255 A in the memory mapped I/O.

xiv) To design and interface a circuit to convert digital data into analog signal using the 8255 A in the memory mapped I/O.

xv) To interface a keyboard with the microprocessor using 8279 chip and transfer the output to the printer.

xvi) To design a circuit to interface a memory chip with microprocessor with given memory map.

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

6th Semester

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course	Code	L	T	P	Credits
Computer Networks	**ITT350**	**3**	**1**	**0**	**4**

Course Outcomes (COs):

CO1: Understand the need for networking in general and computer networks in particular.
CO2: Study the prevalent TCP/IP reference model and understand various error, flow and access control strategies.
CO3: Study the IPV4 addressing and the strategies to delay the transition to IPV6 using techniques like subnetting, VLSM, NAT.
CO4: Understand routing in networks and study various routing algorithms.
CO5: Study the mechanism for connection establishment, termination between the nodes and the ways of reducing network congestion.
CO6: Understand various application layer services.

Syllabus:

UNIT I - INTRODUCTION:
History and development of computer networks, networks topologies. Layering and protocols.

UNIT II - PHYSICAL LAYER:
Different types of transmission media, errors in transmission: attenuation, noise. Encoding (NRZ, NRZI, Manchester, AMI, etc.).

UNIT III - DATA LINK LAYER AND SWITCHING:
MAC Layer: Aloha, CSMA, CSMA/CD, CSMA/CA protocols. Data Link Layer: Error detection (Parity, CRC), Framing, Sliding Window, Stop and Wait protocols, HDLC. Switching Theory: Circuit Switching, Message switching, Packet switching.

UNIT IV - NETWORK LAYER:
Network layer: Internet Protocol, IPv6, ARP, DHCP, ICMP, Routing algorithms: Distance vector,

Link state, Metrics, Inter-domain routing. Dijkstra, Bellman-Ford Algorithms. Subnetting, Classless addressing, Network Address Translation.

UNIT V - TRANSPORT AND APPLICATION LAYER:

Transport layer: UDP, TCP. Connection establishment and termination, sliding window revisited, flow and congestion control, timers, retransmission, TCP extensions.

Application layer: DNS, SMTP, IMAP, HTTP, etc.

Recommended Books:

1. JF Kurose, KW Ross, Computer Networking: A Top-Down Approach.
2. Behrouz A. Forouzan, Data communications and Networking.
3. Andrew S. Tanenbaum, Computer Networks.
4. William Stallings, Data and Computer Communications.

Course	Code	L	T	P	Credits
Artificial Intelligence	**ITT351**	**3**	**0**	**0**	**3**

Course Outcomes (COs):

CO1: Gain historical perspective of AI and its formation.
CO2: Introduce the basic principles of AI problem solving.
CO3: Apply basic principles in problem solving, inference, perception, knowledge representation and learning.
CO4: Investigate applications of AI techniques in intelligent agents, expert systems, machine learning models.
CO5: AI development tools and techniques.

Syllabus:

UNIT I - INTRODUCTION:

Introduction to AI and intelligent agents.Problem Solving : Solving Problems by Searching, heuristic search techniques, constraint satisfaction problems, stochastic search methods, adversarial search,Game playing : minimax, alpha-beta pruning.

UNIT II - KNOWLEDGE REPRESENTATION AND REASONING:

Building a Knowledge Base : Propositional logic, first order logic, Theorem Proving in First Order Logic. Production Systems, Semantic Nets, Frames and Scripts Formalisms. Resolution in Predicate Logic, Unification, Strategies for Resolution by Refutation. Knowledge Acquisition and learning: Learning from examples and analogy, Rote learning, Neural Learning, Integrated Approach. Planning, partial order planning. Uncertain Knowledge and Reasoning, Probabilities, Bayesian Networks.

UNIT III - INTRODUCTION TO MACHINE LEARNING:

Machine Learning, Supervised Learning, Unsupervised Learning, Reinforcement Learning. Introduction to Probability, Basics Linear Algebra, Statistical Decision Theory – Regression & Classification, Bias – Variance, Overfitting and complexity; training, validation, test data.

UNIT IV - EXPERT SYSTEM:

Existing Systems (DENDRAL, MYCIN), domain exploration, Meta Knowledge, Expertise Transfer, Self Explaining System.Fuzzy logic: Fuzzy Logic Propositional logic, Membership functions, Fuzzy logic, Fuzzy rule generation, De-fuzzification, Time dependent fuzzy logic, Temporal fuzzy logics, Case study-to use fuzzy logic for processes control problem
Programming Language: Introduction to programming Language- LISP, PROLOG

UNIT V - NEURAL NETWORKS:

Overview of different forms of learning, Learning Decision Trees, Neural Networks- Basics of Neural Networks: Perceptrons, Feedforward nets Backpropagation algorithm, preliminary understanding of unsupervised learning.
Pattern Recognition: Introduction to Pattern Recognition, Structured Description, Symbolic Description, Machine perception, Line Finding, Interception, Semantic & Model, Object Identification, Speech Recognition.

Text Books:

1. Rich & Knight, “Artificial Intelligence”.
2. Elamie, “artificial Intelligence”, Academic Press.

Reference Books:

1. Char nick “Introduction to Artificial Intelligence”, Addision Wesley.

2. Winston, “LISP”, Addison Wesley.

Course	Code	L	T	P	Credits
Computer Graphics	**ITT352**	**3**	**0**	**0**	**3**

Course Outcomes (COs):

CO1: Understand the basics of computer graphics, different graphics systems and applications of computer graphics.
CO2: Provide an understanding of how to scan convert the basic geometrical primitives, how to transform the shapes to fit them as per the picture definition.
CO3: Use of geometric transformations on graphics objects and their application in composite form.
CO4: Extract scene with different clipping methods and its transformation to graphics display device.
CO5: Explore projections and visible surface detection techniques for display of 3D scene on 2D screen.
CO5: Render projected objects to naturalize the scene in 2D view and use of illumination models for this.

Syllabus:

UNIT I - INTRODUCTION:
computer graphics, Co-ordinate representation, Pixel, Raster Scan & Random Scan methods, color, CRT Raster, scan basics, video basics, interactive devices, graphics input and output devices, mouse, track ball, light pen, digitizer, thumb wheel, raster scan graphics, applications.

UNIT II - Line GENERATION:
Points and lines generation algorithm, DDAlines drawing algorithm, Bresenham's lines drawing algorithm, circle generating algorithm, midpoint circle algorithm,midpoint ellipse generating algorithm, other curves, conic sections, polynomial and spline curves, Pixels addressing, filled-area primitives, scan-line polygon filled algorithms, inside-outside tests, scan-line fill of curved

boundary algorithms,boundary fill algorithms,flood-fill algorithms, fill-area functions, character generation.

UNIT III - SEGMENTS:
Segments table, Creating, Deleting and renaming a segment Visibility, Image transformation.

UNIT IV - TRANSFORMATION:
2D Transformation, An introduction to 3D transformation, Projections, Light, color and shading.

UNIT V - WINDOWING AND CLIPPING:
Viewing transformation, Clipping. Generalized clipping IN 2D.
Hidden line and surfaces: Back-face Removal Algorithms, Hidden line methods
Rendering and Illumination: Introduction to curve generation, Bezier, Hermite and B-spline algorithms and their Comparisons

Text Books:
1. Computer Graphics (Principles and Practice) by Foley, van Dam, Feinerand Hughes, Addisen Wesley.
2. Computer Graphics by D Hearn and P M Baker, Printice Hall of India.
3. Mathematical Elements for Computer Graphics by D F Rogers, McGraw Hill
4. Procedural Elements for Computer Graphics by D F Rogers, McGraw Hill

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course	Code	L	T	P	Credits
Big Data	**ITT353**	**3**	**0**	**2**	**4**

Course Outcomes (COs):

CO 1: Understand Big Data, its platform and its use cases .
CO 2: Provide an overview of Apache Hadoop ,HDFS Concepts and Interfacing with HDFS
CO 3: Understanding Data Sciences and Data life cycle
CO 4: Understanding and Using Supervised and Unsupervised Learning Algorithms
CO 5: Tools and Technologies for Unstructured Data Analytics
CO 6: Implementing Machine Learning algorithms using Python

Syllabus:

Unit I - INTRODUCTION:

Big Data Overview, Introduction to the Big Data problem. Current challenges, trends, and applications, Algorithms for Big Data analysis. Data sets, Mining and learning algorithms that deal with large datasets Technologies for Big Data management. Big Data technology and tools, special consideration made to the Map-Reduce paradigm and the Hadoop ecosystem.

Unit II - DATA SCIENCE:

What is data sciences, The rising and importance of data sciences, Big data analytics in industry verticals, Data Analytics Lifecycle and methodology, Data Understanding, Data Preparation.

Unit III - MODELING:

Evaluation, Communicating results, Deployment, Data exploration & preprocessing.

Unit IV - MEASURES AND EVALUATION:
Data Analytics: Theory & Methods, Supervised learning, Linear/Logistic regression, Decision trees, Naïve Bayes, Unsupervised learning, K-means clustering, Association rules

Unit V - UNSTRUCTURED DATA ANALYTICS:
Technologies & tools, Text mining, Web mining, Operationalizing an Analytics project, Data Visualization Techniques, Creating final deliverables
Term project: Using Amazon AWS, BlueMix, Cognos, Biginsights.

Text Books:

1. Big Data: A Revolution That Will Transform How We Live, Work, and Think by Viktor Mayer-Schönberger, Kenneth Cukier.
2. Hadoop: The Definitive Guide by Tom White (Goodreads Author), Doug Cutting , oreily Publiactions.
3. Real-Time Big Data Analytics: Emerging Architecture [Kindle Edition], Mike Barlow

Course	Code	L	T	P	Credits
Object-Oriented Programming II with Java	**ITT354**	**3**	**0**	**0**	**3**

Course Outcomes (COs):

CO1: Understand various basics related to java programming, object-oriented programming and other concepts like JVM, JVM architecture, JIT compilation.

CO2: Understand the underlying principles of object-oriented programming like abstraction, polymorphism etc and getting familiar with various java classes. Learn to define and import packages, implement interfaces.

CO3: Getting familiar with exception and string handling in java. Study creation, concatenation and conversion of a string, searching and modification, string comparison. String Buffer and StringBuilder classes and Date class.

CO4: Have detailed knowledge on concurrent programming and file handling. Study various data collectors available in java like ArrayList, LinkedList, Queue etc. Understanding thread execution, multithreading, thread priorities and scheduling, synchronization. Understanding file handling, creating, writing, reading, updating, touching and deleting files, Byte Streams and Character Streams, InputStream & OutputStream classes and their subclasses, Reader and Writer classes and their subclasses.

CO5: Be familiar with graphical components like buttons, labels, events, windows etc through which user can interact with the applications. Working with controls and layout managers, event handling and data validation.

Syllabus:

UNIT I - INTRODUCTION:

What is Java? Background/History of Java, Java Virtual Machine, JVM Architecture, Byte code, HotSpot JVM and JIT Compilation, Basics of OOP. Introduction to Classes and Objects. Data types. Garbage collection: Eden space, Survivor Space, Tenured generation, Permanent generation, Code cache, loops and flow control.

UNIT II - OBJECT ORIENTED PROGRAMMING CONCEPTS:

Abstraction, Encapsulation, Polymorphism and Overloading, Constructors and destructors scope of declarations, Access Control, Nested and Inner classes. Array handling. Using extends keyword, subclass, super-class, over- riding methods, dynamic method dispatch, The Object class, Abstract and final classes. Packages: defining, importing, Access Control. Interface: Defining, Implementing and applying interface. Wrapper classes.

UNIT III - EXCEPTION AND STRING HANDLING:

Basic exceptions, user defined exceptions, catching exceptions – try, catch and multi try catch, throwing and re-throwing, finally clause. String Handling: Creation, concatenation and conversion of a string, searching and modification, string comparison. StringBuffer and StringBuilder classes and Date class.

UNIT IV - CONCURRENT PROGRAMMING AND FILE HANDLING:

Generics & Collections: List interface, ArrayList, LinkedList, Queue, Stack,
Threads: Create new threads – extending java.lang.Thread, implementing java.lang.Runnable Interface, Understanding thread execution, multithreading, thread priorities and scheduling, synchronization Introduction to java.util.concurrent classes and interface and using java.util.concurrent.Callable interface. Introduction to Fork-Join Framework.
File handling, Creating, writing, reading, updating, touching and deleting files, Byte Streams and Character Streams, InputStream & OutputStream classes and their subclasses, Reader and Writer classes and their subclasses.

UNIT V - GUI COMPONENTS:

Introduction to AWT and Swing, frames, panels, buttons and events, layout managers, text fields, labels. Working with controls and layout managers, event handling and data validation, Applets. Introduction to JavaFx.

Text Books:

1. Java for Programmers, P.J. Dietel, H. M. Dietel, Pearson Education.
2. Java SE 6, Joel Murach, A. Steelman, SPD Pvt. Ltd.
3. Head first java, Kathy Sierra, Bert Bates, Oreilly.
4. Core Java, Cay Horstman and Gary Cornell, Prentice Hall

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY SRINAGAR

Course	Code	L	T	P	Credits
Computer Networks Lab	**ITL355**	**0**	**0**	**2**	**1**

Course Outcomes (COs):

CO1: Understand colour coding of guided media and create crossover and straight through cable.
CO2: Implement basic network utilities and analyse network traffic using Wireshark tool.
CO3: Hands-on Cisco Packet Tracer by building basic networks and configuring internetworking devices like router, switch.
CO4: Implement static and dynamic routing in Packet Tracer and configure access control lists.
CO5: Simulate wireless networks using NS3.

Syllabus:

1. Study and implementation of colour coding standards of guided media (UTP).
2. Implementation and understanding of basic network utilities: ping, ipconfig/ifconfig, mstsc, nslookup, tracert.
3. Network traffic capture and analysis using Wireshark.
4. Introduction to Cisco Packet Tracer: Building a LAN with HUBs and Switches, understand and implement address learning in switches.
5. Router Configuration Using Packet Tracer, IP addressing (static and dynamic), subnetting.

6. Static/Dynamic Route Configuration.
7. Implementation of routing protocols (RIP, OSPF, BGP).
8. Standard access control list (ACL) configuration, Extended access control list (ACL) configuration.
9. Implementation of flow control protocols.

Course	Code	L	T	P	Credits
Artificial Intelligence Lab	**ITL356**	**0**	**0**	**2**	**1**

Course Outcomes (COs):

CO1: Understand simple facts and variables.
CO2: Implement and apply simple predicates, predicate inference and goal queries
CO3: Demonstrate the proficiency in applying scientific methods to models of machine learning.

Syllabus:

AI PYTHON LAB CONTENTS

1. Input & Output
2. Operators and Arithmetic
3. Facts & Variables
4. Simple facts and facts with arguments
5. Rules & Predicates
6. Simple Predicates, Predicate Inference, Goal queries
7. Recursion
8. Graph Traversal
9. Depth First Search, Breadth First Search

Course	Code	L	T	P	Credits
Computer Graphics Lab	**ITL357**	**0**	**0**	**2**	**1**

Course Outcomes (COs):

CO1: Understand the basic concepts of computer graphics.
CO2: Design scan conversion problems using C programming.
CO3: Apply clipping and filling techniques for modifying an object.
CO4: Understand the concepts of different type of geometric transformation of objects in 2D and 3D.
CO5: Understand the practical implementation of modeling, rendering, viewing of objects in 2D

Syllabus:

1. Point drawing to understand co-ordinate system of display device.
2. To implement Bresenham's algorithms for line generation.
3. To implement DDA algorithm for line generation.
4. To implement midpoint circle generation algorithm
5. To implement midpoint ellipse generation algorithm
6. To implement flood-fill and boundary fill algorithm.
7. To perform 2D Transformations such as translation,
8. To perform 2D Transformations such as rotation,
9. To perform 2D Transformations such as scaling,
10. To perform 2D Transformations such as reflection
11. To perform 2D Transformations such as shearing.
12. To implement Cohen-Sutherland 2D clipping and window-viewport mapping.
13. To perform 3D Transformations such as translation, rotation and scaling.

14. To visualize projections of 3D images.
15. To convert between color models.
16. To implement text compression algorithm using librarires.
17. To implement image compression algorithm using librarires.
18. To perform animation using any Animation software.
19. To perform basic operations on image using any image editing software.

20. Implementation of viewing/rendering pipeline.

Course	Code	L	T	P	Credits
Object-Oriented Programming II with Java Lab	**ITL358**	**0**	**0**	**2**	**1**

Course Outcomes (COs):

CO1: Implementing the underlying principles of object-oriented programming like abstraction, polymorphism etc and various java classes.
CO2: Learn to define and import packages, implement interfaces.
CO3: Implementing exception handling, creating user defined exceptions, catching exceptions. Creation, concatenation and conversion of a string, searching and modification, string comparison. Implementing String Buffer and StringBuilder classes and Date class.
CO4: Using various data collectors available in java like ArrayList, LinkedList, Queue etc. Implementing multithreading, thread priorities and scheduling, synchronization. Implementing various file handling, creating, writing, reading, updating, touching and deleting files, Byte Streams and Character Streams, InputStream & OutputStream classes and their subclasses, Reader and Writer classes and their subclasses.
CO5: Executing graphical components like buttons, labels, events, windows etc through which user can interact with the applications.

Syllabus:

1. Java package with simple stack and queue class

2. Complex number manipulation

3. Date class similar to java.util package
4. Implementing dynamic polymorphism in java
5. Java interface for ADT stack
6. Developing a simple paint like program using applet
7. Developing a scientific calculator
8. Developing a template for linked list
9. Develop a multi threaded producer consumer Application
10. Generating prime numbers and Fibonacci series
11. Multithreaded GUI application