Semester 3

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: BSCS101

Course Title: Applied Mathematics

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 0	**Credits:** 4
Semester: 3	**Theory/Practical:** Theory	**Teaching Hours:** 45(L) + 15(T) = 60 hrs
Total Max. Marks: 100	**Continuous Assessment (CA) Marks:** 40	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 95%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): BSC102, BSC104

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply concepts of inner product spaces to solve problems involving orthogonalization Gram Schimdt process.
2	Use Laplace transforms to solve ordinary differential equations with initial conditions.
3	Apply Fourier integrals to solve problems in engineering.
4	Implement interpolation algorithm in computation tools for engineering applications.
5	Implement numerical techniques for solving algebraic equations and for performing numerical integration.

Contents

Part-A

Unit-1 Laplace Transforms **11 (L)hrs**

Definition and existence of Laplace Transforms, Laplace transforms of various standard functions, properties of Laplace transforms, inverse Laplace transforms, transform of derivatives and integrals, Transform of multiplication and division by t, convolution theorem, Laplace transform of unit step function. Applications to solution of ordinary linear differential equations with constant coefficients, Applications to solve initial and boundary value problems.

Unit-2 Fourier Series **12 (L)hrs**

Introduction, even and odd functions, periodic functions, Dirichlet's conditions for Fourier series, Euler's formulae for Fourier series expansion, change of interval, Half Range series expansions, Fourier series on arbitrary intervals, Application of Fourier series to solve different waveforms.

Part-B

Unit-3 Numerical Methods **10 (L)hrs**

Bisection Method, Newton Raphson Method to solve the algebraic equations, Simpon's Method and Trapezoidal method to evaluate the definite integral, interpolation, forward and backward differences, Newton's Forward Interpolation Formula and Newton's Backward Interpolation formula.

Unit-4 Advanced Linear Algebra **12 (L)hrs**

Solution of system of linear equations, Linear Spaces, Vector Spaces, Vector Sub Spaces, Basis and dimension, Inner product, Gram Schmidt orthogonalization process, Eigen-values, Eigen-vectors and Diagonalisation, Projection matrix, orthogonal matrix, idempotent Matrix, Partition matrix, Nullity, LU decomposition.

Tutorials may be conducted in lab settings and SageMath/ GeoGebra/ GNU Octave could be used to execute the lessons (as the case may be).

Text Books

1. B.S. Grewal, "Higher Engineering Mathematics", 44th edition, Khanna Publishers, 2021.
2. M.K. Jain, S.R.K. Iyengar, R.K. Jain, "Numerical Methods for Scientific and Engineering Computation", 6th edition, Newage International Publishers, 2003.

Reference Books

1. Erwin Kreyszig, "Advanced Engineering Mathematics", 9th edition, Wiley India Pvt Ltd, 2016.
2. B.S. Grewal, "Numerical methods in Engineering and Science with Programes in C, C++ & Matlab", 11th, Khanna Publishers, 2013
3. Axler, Sheldon, "Linear Algebra Done Right", 4th, Cham, Switzerland: Springer International Publishing, 2024.
4. Iyengar, T.K.V., B.Krishna Gandhi and S. Ranganatham & M.V.S.S.N. Prasad, "Laplace Transforms, Numerical Methods & Complex Variables", S. Chand Publishing, 2018.
5. Sreenadh, S, "Fourier Series and Integral Transforms", New Delhi, India: S Chand, 2014.
6. Atkinson, Kendall, Nashville, TN "An Introduction to Numerical Analysis", 2nd edition, John Wiley & Sons, 1989.
7. Strang, Gilbert, "Introduction to Linear Algebra", 6th edition, Wellesley, MA: Wellesley-Cambridge Press, 2023.

Online Learning Materials

1. https://www.cimt.org.uk/projects/mepres/alevel/fstats_ch5.pdf

Accessed on April 12,2025

2. https://www.cs.ox.ac.uk/files/12921/book.pdf
   Accessed on April 14,2025
3. https://www.youtube.com/watch?v=nH05UiErAX4
   Accessed on April 12,2025
4. https://www.youtube.com/watch?v=d7NF-_8vVv4&t=2s
   Accessed on April 14,2025
5. https://www.webpages.uidaho.edu/~stevel/519/Applied%20Multivariate%20Statistical%20Analysis%20by%20Johnson%20and%20Wichern.pdf
   Accessed on April 14,2025
6. http://ndl.ethernet.edu.et/bitstream/123456789/55096/1/Tsuneo%20Arakawa.pdf
   Accessed on April 14,2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Linear Algebra	Prof. Pranav Haridas	Kerala School of Mathematics	https://onlinecourses.nptel.ac.in/noc25_ma37/preview
2	Numerical Analysis	Prof. S. Baskar	IIT Bombay	https://onlinecourses.nptel.ac.in/noc25_ma41/preview
3	Fourier Analysis and its applications	Prof. Gopala Krishna Srinivasan	IIT Bombay	https://onlinecourses.nptel.ac.in/noc25_ma23/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS101
Course Title: Object Oriented Programming

Programme: B.Tech.	**L:** 2 **T:** 0 **P:** 4	**Credits:** 4
Semester: 3	**Theory/Practical:** Theory	**Teaching Hours:** 30(L) + 60 (P) = 90 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): ESC103

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Demonstrate programming skills in writing, compiling and debugging C++ programs.
2	Use class and object to implement object-oriented approach.
3	Apply inheritance and polymorphism to implement object-oriented programs.
4	Analyse the relevance of dynamic memory management for effective memory utilization.
5	Make use of exception handling and file handling in program development.
6	Develop solution for programming problems using object-oriented principles.

Contents

Part-A

Unit-1 Object Oriented Programming Concepts **4(L) hrs**

Introduction, Comparison between Procedure oriented approach and Object-oriented approach, Basic data types, Derived data types, Keywords, Identifiers, Constants and variables, Type casting, Operators, and Operator precedence. Control Structures- if statement, switch-case, for, while and do-while loops, break and continue statement. Features of object-oriented programming- Class, Objects, Encapsulation, Abstraction, Data hiding, Polymorphism and Inheritance.

Unit-2 Classes and Objects **5(L) hrs**

Implementation of a class, Class objects, Access specifiers, Accessing class members, Constructor and destructor, Types of constructors, Empty class, Nested class, Friend function, Inline function, Friend class, Static members.

Unit-3 Inheritance **5(L) hrs**

Introduction, Defining derived classes, Types of inheritance, Ambiguity in inheritance, Virtual base class, Objects slicing, Overriding member functions.

Part-B

Unit -4 Dynamic Memory Management using Pointers **5(L) hrs**

Declaring and initializing pointers, Accessing data through pointers, Pointer arithmetic, Memory allocation – Comparison of Static and Dynamic, Dynamic memory management using new and delete operators, Pointer to an object, this pointer, Pointer related problems – Dangling/wild pointers, Null pointer assignment, Memory leak and Allocation failures.

Unit-5 Polymorphism and Type Conversion **6(L) hrs**

Concept of binding – Early binding and late binding, Virtual functions, Pure virtual functions, Abstract class. Function overloading, Constructor overloading, Operator Overloading, Rules for overloading operators, Overloading of various operators. Type conversion – Basic type to class type, Class type to basic type, Class type to another class type.

Unit-6 Exceptions Handling and File Handling **5(L)hrs**

Review of traditional error handling, Basics of exception handling, Exception handling mechanism, Throwing and catching mechanism, Rethrowing an exception, Specifying exceptions. File streams, Error handling during file operations, Reading/writing of files, accessing records randomly, Updating files.

Laboratory Work

In the laboratory work, students will implement programs related to above topics using any open-source C++ tools, such as Code Blocks, Dev-C++, VS Code etc.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Program to illustrate the use of class and object.
2.	Program to use decision control statements using class and object.
3.	Program to use Loop control statements using class and object.
4.	Program to illustrate the concept of nesting of member functions.
5.	Program to illustrate the concept of default parameters in a function.
6.	Program to implement a local class.
7.	Program to illustrate the concept of nested class.
8.	Program to illustrate the concept of inline function.
9.	Program to illustrate the concept of friend function.
10.	Program to illustrate the concept of empty class.
11.	Program to show the working of static data members in a class.
12.	Program to show the working of static functions in a class.

13.	Program to illustrate the concept of friend class.
14.	Program to illustrate the concept of default constructor.
15.	Program to illustrate the use of parameterized constructor.
16.	Program to illustrate the use of copy constructor.
17.	Program to illustrate the concept of dynamic constructor.
18.	Program to illustrate the concept of destructors.
19.	Program to illustrate the concept of single inheritance.
20.	Program to illustrate the concept of multilevel inheritance.
21.	Program to illustrate the concept of multiple inheritance.
22.	Program to illustrate the concept of hierarchical inheritance.
23.	Program to illustrate the concept of hybrid inheritance.
24.	Program to illustrate the concept of ambiguity in single inheritance.
25.	Program to illustrate the concept of ambiguity in multiple inheritance.
26.	Program to illustrate the concept of virtual base class/multipath inheritance.
27.	Program to illustrate the concept of dynamic memory management using new and delete.
28.	Program to illustrate the use of this pointer.
29.	Program to show pointer arithmetic operations.
30.	Program to overload unary minus operator.
31.	Program to overload prefix and postfix increment operators.
32.	Program to overload binary addition operator.
33.	Program to add two complex numbers using binary operator overloading.
34.	Program to illustrate the use of virtual function.
35.	Program to illustrate the concept of abstract class.
36.	Program to illustrate basic to class type conversion.
37.	Program to illustrate class to basic type conversion.
38.	Program to illustrate class to class type conversion using constructor.
39.	Program to illustrate class to class type conversion using casting operator.
40.	Program to illustrate the concept of exception handling for throw point inside the try block.
41.	Program to illustrate the concept of exception handling for throw point outside the try block.
42.	Program to illustrate the concept of exception handling for multiple catch statements.
43.	Program to illustrate the concept of exception handling for rethrowing an exception.
44.	Program to illustrate the concept of exception handling with single catch statement for all exceptions.
45.	Program to perform read and write operations on file using get(), put().

46.	Program to perform read and write operations on file using read(), write().
47.	Program to illustrate the concept of file pointers.
48.	Any one project i. Banking System: The Banking System project has account class with data members like account number, name, deposit, withdraw amount and type of account. Customer data is stored in a binary file. A customer can deposit and withdraw amount in his account. User can create, modify and delete account. ii. Library Management System: Creating a simple text-based system for managing library. It allows users (admin/librarian) to add new books, view book details, and update availability status. Users should be able to issue books to students (marking them as issued) and return them. It also maintains details such as book title, author, ID and availability. Implement using file handling to preserve data like issued books and book inventory even after closing the application. iii. Student Grade Management System: Design console-based application that allows the user (e.g., a teacher) to enter student names, roll numbers, and their marks for various subjects. The program will then calculate total marks, average percentage, and assign a grade (A, B, C, etc.) based on predefined criteria. It should also be able to display the individual student's report card and allow editing or viewing student records. Add file handling to save and retrieve student data between sessions. iv. Simple ATM Interface: This project simulates a basic ATM interface that requires users to enter a PIN for authentication. Once logged in, the user can perform operations such as checking balance, withdrawing cash and depositing money. The account details (like current balance) should be stored and updated after every transaction. It should work for multiple user accounts and use file handling to save data persistently. Any other project recommended by the instructor.

Text Books

1. E. Balagurusamy, “Object Oriented Programming with C++”, 8th edition, Tata McGraw Hill, 2020.
2. P Yashavant Kanetkar, “Let Us C++”, 3rd edition, BPB Publications, 2019.

Reference Books

1. Paul Deitel and Harvey Deitel, “C++ How to Program”, 10th edition, Pearson, 2016.
2. R. Lafore, “Object Oriented Programming in C++”, 4th edition, Waite Group, 2002.
3. Herbert Schildt, “The Complete Reference to C++ Language”, 4th edition, McGraw Hill-Osborne, 2017.

Online Learning Materials

1. https://faculty.iitr.ac.in/cs/bala/OOPS/csn103.html Accessed on 24-04-2025
2. https://www.youtube.com/watch?v=iw1Xf_33YM0&list=PLQEaRBV9gAFujcBWJhBT2XXsuMlIfETBy Accessed on 24-04-2025
3. https://www.youtube.com/watch?v=qq3BY4viEB4&list=PLQEaRBV9gAFujcBWJhBT2XXsuMlIfETBy&index=4 Accessed on 24-04-2025
4. https://www.youtube.com/watch?v=p2h8rGnkD0o&list=PLQEaRBV9gAFujcBWJhBT2XXsuMlIfETBy&index=6 Accessed on 24-04-2025
5. https://www.youtube.com/watch?v=NBsmPHXjLfg&list=PLQEaRBV9gAFujcBWJhBT2XXsuMlIfETBy&index=8 Accessed on 24-04-2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Fundamentals of Object-Oriented Programming	Prof. Balasubramanian Raman	IIT Roorkee	https://onlinecourses.nptel.ac.in/noc25_cs34/preview
2	Programming in C++	Prof. Partha Pratim Das	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc20_cs07/preview
3	Programming in Modern C++	Prof. Partha Pratim Das	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc25_cs58/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS102
Course Title: Data Structures

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 4	**Credits:** 6
Semester: 3	**Theory/Practical:** Theory	**Teaching Hours:** 45(L) + 15(T) + 60(P) = 120 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 50%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): ESC103

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply Arrays and Linked Lists to solve computational problems.
2	Implement stack and queue data structures to solve programming problems.
3	Apply appropriate indexing and searching techniques to optimize data storage and retrieval.
4	Analyze non-linear data structures to perform efficient storage, retrieval, and traversal operations.
5	Analyze time and space complexity of different sorting algorithms.
6	Identify the appropriate data structure to provide solution with reduced space and time complexity.

Contents

Part-A

Unit-1 Basic concepts and Arrays **5(L) hrs**

Concept of data type, Linear and non-linear data structures, Data structures versus data types, Operations on data structures, Algorithm complexity and Asymptotic notations. Linear and multi-dimensional arrays and their representation, Operations on arrays, Sparse matrices and their storage.

Unit-2 Searching and Sorting **4(L) hrs**

Linear and binary search techniques, Sorting methods – Bubble sort, Selection sort, Insertion sort, Merge sort, Shell sort and radix sort. Complexities of searching and sorting algorithms.

Unit -3 Linked List **7(L) hrs**

Definition and representation of Linked list, Types of Linked list- Linear linked list, Doubly linked list, Circular linked list and Header linked list and their operations, Application of linked lists.

Unit -4 Stacks **7(L) hrs**

Sequential and Linked representation of stacks, Operations on stacks, Application of stacks – parenthesis checker, Evaluation of postfix expressions, Conversion from infix to postfix, Conversion from infix to prefix representation, Tower of Hanoi problem, implementing recursive functions, Quick sort.

Part-B

Unit-5 Queues **4(L) hrs**

Sequential and Linked representation of queue, Types of queue- Linear Queue, Circular Queue, Dequeue, Priority Queue, Operations on each types of Queues and their algorithms, Applications of Queues.

Unit-6 Trees **7(L)hrs**

Basic terminology, Sequential and linked representations of trees, Tree Traversals, Different types of Trees- Binary Tree, Binary search tree, Threaded binary tree, AVL tree, B-tree, B^+- tree, Red Black tree. Operations on each of the trees. Application of Binary Trees.

Unit-7 Heaps **2(L)hrs**

Representing a heap in memory, Operations on heaps, Application of heap in implementing priority queue and Heap sort algorithm.

Unit-8 Graphs **6(L)hrs**

Basic terminology, Representation of graphs – Adjacency matrix, Adjacency list. Operations on graph, Traversal of a graph – Breadth first search, Depth first search. Shortest path algorithms -Dijkstra's and Floyd-Warshall algorithm. Minimum spanning tree – Prim and Kruskal. Applications of graphs.

Unit-9 Hashing and Hash Tables **3(L)hrs**

Introduction to hash table, Hash functions, Concept of collision and its resolution using open addressing and separate chaining, Double hashing, Rehashing.

Tutorial hours will be used for practice sessions for numerical/programming problems and design of algorithms related to above contents.

Laboratory Work

In the laboratory work, students should implement these data structures using any open-source C++ tools, such as Code Blocks, Dev-C++, VS Code etc.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Write a program to add two matrices.
2	Design, Develop and Implement a menu driven Program for the following Array operations a. Display of Array Elements with Suitable Headings b. Inserting an Element (ELEM) at a given valid Position (POS) c. Deleting an Element at a given valid Position (POS) d. Exit.
3	Write a Program to find the position of an element in an array using Linear Search Algorithm.
4	Write a Program to find the position of an element in an array using Binary search Algorithm.
5	Write a program to sort list using bubble sort.
6	Write a program to sort list using selection sort.
7	Write a program to sort list using insertion sort.
8	Write a program to sort list using merge sort.
9	Design, Develop and Implement a menu driven Program for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Branch, Sem, Phone Number a. Insert new student at beginning of list. b. Insert new student at end of list. c. Insert new student in the middle of the list. d. Display all records and count the number of nodes in it. e. Delete first student from the list. f. Delete last student from the list g. Delete nth student from the list. h. Exit.
10	Design, Develop and Implement a Program to perform insertion, deletion and traversing in a sorted Singly Circular Linked List (SCLL) with header nodes.

Experiment No.	Experiment Title
11	Design, Develop and Implement a menu driven Program for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, Phone Number a. Insert new student at beginning of list. b. Insert new student at end of list. c. Insert new student in the middle of the list. d. Display all records and count the number of nodes in it. e. Delete first student from the list. f. Delete last student from the list g. Delete nth student from the list. h. Exit.
12	Implement a Program to find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z)
13	Design, Develop and Implement a menu driven Program for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) a. Push an Element on to Stack b. Pop an Element from Stack d. Display the elements of Stack e. Exit
14	Design, Develop and Implement a menu driven Program for the following operations on STACK of Integers (Linked Implementation) a. Push an Element on to Stack b. Pop an Element from Stack d. Display the elements of Stack e. Exit
15	Design and implement a program to check if a given number is Palindrome or not using stack.
16	Design, Develop and Implement a Program for evaluating Suffix expression with single digit operands and operators: +, -, *, /, %, ^.
17	Design, Develop and Implement a Program for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, % (Remainder), ^(Power) and alphanumeric operands
18	Design, Develop and Implement a Program for solving Tower of Hanoi problem with n disks
19	Write a program to sort list using quicksort.

Experiment No.	Experiment Title
20	Design, Develop and Implement a menu driven Program for the following operations on Linear QUEUE (Array Implementation of Queue with maximum size MAX) Insert an Element on to Linear QUEUE Delete an Element from Linear QUEUE Display the elements of Linear QUEUE Exit Support the program with appropriate functions for each of the above operations.
21	Design, Develop and Implement a menu driven Program for the following operations on QUEUE (Linked Implementation) Insert an Element on to QUEUE Delete an Element from QUEUE Display the elements of QUEUE Exit Support the program with appropriate functions for each of the above operations.
22	Design, Develop and Implement a menu driven Program for the following operations on Circular QUEUE (Array Implementation of Queue with maximum size MAX) a. Check Overflow and Underflow situations on Circular QUEUE b. Insert an Element on to Circular QUEUE c. Delete an Element from Circular QUEUE d. Display the elements of Circular QUEUE e. Exit Support the program with appropriate functions for each of the above operations
23	Design, Develop and Implement a menu driven Program for the following operations on Dequeue QUEUE (Array Implementation of Queue with maximum size MAX) a. Check Overflow and Underflow situations on Dequeue QUEUE b. Insert an Element on to Dequeue QUEUE c. Delete an Element from Dequeue QUEUE d. Display the status of Dequeue QUEUE e. Exit Support the program with appropriate functions for each of the above operations

Experiment No.	Experiment Title
24	Design, Develop and Implement a menu driven Program for the following operations on Priority QUEUE (Array Implementation of Queue with maximum size MAX) Insert an Element on to Priority QUEUE Delete an Element from Priority QUEUE Display the status of Priority QUEUE Exit Support the program with appropriate functions for each of the above operations
25	Design, Develop and Implement a menu driven Program for the following operations on Binary Search Tree (BST) of Integers. a. Insertion in a BST. b. Traverse the BST in Inorder, Preorder and Post Order. c. Search an element in BST d. Deletion in BST e. Exit
26	Design, Develop and Implement a menu driven Program for the following operations on AVL Tree of Integers. Insertion in an AVL Tree. Search an element in AVL Deletion in AVL Exit
27	Write a program to sort list using Heapsort.
28	Design, Develop and Implement a menu driven Program for the following operations on Graph(G) of Cities a. Create a Graph of N cities using Adjacency Matrix b. Traverse Graph using DFS c. Traverse Graph using BFS method
29	Design, Develop and Implement a program to construct spanning tree using Kruskal's and Prims algorithm
30	Design, Develop and Implement a program to find shortest path using Dijkstra's algorithm.
31	Construct a program to implement Hash Functions and Collision Resolution Technique

Students are required to make one mini-project based on above data structures.

Text Books

1. Seymour Lipschutz, "Schaum's Outline of Data Structures", McGraw-Hill Education, 1st edition, 1986.
2. Sartaj Sahni, "Data Structures, Algorithms, and Applications in C++", 2nd edition, Silicon Press, 2004.

Reference Books

1. Michael T. Goodrich, Roberto Tamassia, and David M. Mount, "Data Structures and Algorithms in C++", 2nd Edition, Wiley, 2011.
2. Robert L. Kruse, "Data Structures and Program Design in C", 2nd Edition, Prentice Hall, 1996.
3. Yedidyah Langsam, Moshe J. Augenstein, and Aaron M. Tenenbaum, "Data Structures Using C and C++", 2nd edition, Prentice Hall, 1995.
4. Vishal Goyal, Lalit Goyal and Pawan Kumar, "Simplified Approach to Data Structures", 1st edition, Shroff Publishers, 2014.

E-Books and Online Learning Material

1. Data Structures and Algorithms: by Granville Barnett, and Luca Del Tongo.
   https://www.mta.ca/~rrosebru/oldcourse/263114/Dsa.pdf Accessed on 15-05-2025

2. Data Structures and Algorithms in JAVA: by Michael T. Goodrich and Roberto Scheme and Tamassia
   https://eduarmandov.wordpress.com/wp-content/uploads/2017/05/c_c-data-structures-and-algorithms-in-c.pdf Accessed on 15-05-2025

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106102064 Accessed on 15-05-2025
2. https://nptel.ac.in/courses/106106127 Accessed on 15-05-2025
3. https://nptel.ac.in/courses/106105085 Accessed on 15-05-2025
4. https://nptel.ac.in/courses/106106133 Accessed on 15-05-2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Data Structures	Dr. S. Sasikala	Institute of distance Education University of Madras	https://onlinecourses.swayam2.ac.in/cec19_cs04/preview

2	Data Structures	Dr. M. Deivamani,	College of Engineering, Guindy, (CEG), Anna University	https://onlinecourses.swayam2.ac.in/cec25_ma15/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS103
Course Title: Computer Architecture

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 0	**Credits:** 4
Semester: 3	**Theory/Practical:** Theory	**Teaching Hours:** 45(L) + 15(T) = 60 hrs
Total Max. Marks: 100	**Continuous Assessment (CA) Marks:** 40	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand the Data representation, Register Transfer and micro-operations in computer system.
2	Illustrate the CPU architecture and I/O interfacing for different modes of transfer.
3	Analyze memory hierarchy in computer system for efficient memory management.
4	Understand the organization of components of a computer system and their interconnectivity.
5	Develop assembly language programs using 8085 instruction set to perform fundamental operations.
6	Interface peripheral devices with microprocessors using 8255, 8259, and 8237.

Contents

Part-A

Unit-1 Data Representation **4(L) hrs**

Data types, Complements, Fixed point representation, IEEE 754 Floating point representation (32bit/64bit), Error detection and correction.

Unit-2 Register Transfer and Micro-operations **8(L) hrs**

Addition, Subtraction, Multiplication and division algorithms and hardware, Three-state buffer, Binary Adder, Binary Incrementor, Register-transfer language and operations, Arithmetic micro-operations, Logic micro-operations, Shift micro-operations, Arithmetic logic shift unit**.**

Unit-3 Computer Organization and Design **8(L) hrs**

Instruction codes - Direct and Indirect Address, Computer registers, Computer instructions, Timing and control, Instruction cycle, Instruction format, Memory, register, and input-output reference instructions, Input/ Output and interrupts, Design and working of a complete basic computer, Control functions, Design of accumulator logic.

Unit-4 Memory Organization **4(L) hrs**

Memory hierarchy, High-speed memories, Main Memory, Cache memory, Associative memory, Memory management techniques.

Part-B

Unit-5 Central Processing Unit and Input-Output Organization **8(L) hrs**

General register organization, Stack organization, RISC and CISC architecture, I/O interface and types, Modes of transfer, DMA.

Unit-6 8085 Microprocessor Architecture and Programming **10(L) hrs**

Introduction to microprocessors, 8085 microprocessor architecture Pin Diagram, Bus structure, Addressing modes, Instruction classification, Instruction formats, Data transfer operations, Arithmetic operations, Logical operations, Branch operations, Stack and subroutine operations.

Unit-7 Interfacing **3(L) hrs**

8255 Programmable Peripheral Interface, 8259 interrupt controller and 8237 DMA controller.

Tutorial hours will be used for practical sessions of Assembly Language Programming of 8085 microprocessor using 8085 trainer kits and practice problems related to the course.

Text Books

1. M. Morris Mano, "Computer System Architecture", 3rd edition, Pearson Education, 1993.
2. William Stallings, "Computer Organization and Architecture", 11th edition, Pearson Education, 2019.
3. D.A. Patterson and J.L. Hennessy, "Computer Architecture: A Quantitative Approach", 5th edition, Morgan Koffman, 2011.
4. Ramesh S. Gaonkar, "Microprocessor Architecture, Programming, and Applications with 8085", 6th edition, Penram International Publication, Prentice Hall, 2002.

Reference Books

1. D.A. Patterson and J.L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", 5th edition, Elsevier India, 2016.
2. C. Hamacher, Z. Vranesic and S. Zaky, "Computer Organization", 5th edition, McGraw Hill, 2011.
3. B. Ram, "Fundamentals of Microprocessors and Microcomputers", 7th edition, Dhanpat Rai Publications, 2011.

Online Learning Materials

1. https://www.cse.iitd.ac.in/~srsarangi/archbooksoft.html Accessed on March 20, 2025.
2. https://www.youtube.com/watch?v=UpU5svo87Oo Accessed on March 20, 2025.
3. https://en.wikipedia.org/wiki/Computer_architecture Accessed on March 20, 2025.
4. https://www.coursera.org/learn/comparch Accessed on March 20, 2025.

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1.	Computer Architecture	By Prof. Smruti Ranjan Sarangi	IIT Delhi	https://onlinecourses.nptel.ac.in/noc23_cs67/preview
2.	Microprocessors and Microcontrollers	Prof. Santanu Chattopadhyay	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc25_ee49/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ESCS101
Course Title: Digital Electronics

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 3	**Theory/Practical:** Theory	**Teaching Hours:** 45 (L) + 30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 50%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): ESC101

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply the structure of number systems in digital design.
2	Minimize the Boolean expressions in SOP and POS form using K-maps.
3	Apply knowledge of basic characteristics of logic families to design digital circuits.
4	Demonstrate the basic knowledge of behavioral objects of VHDL in programming.
5	Implement combinational logic circuits using Boolean algebra and logic gates and simulate them using VHDL.
6	Analyze Synchronous and Asynchronous sequential circuits using Flip Flops, registers and Counters and simulate them using VHDL.

Contents

Part-A

Unit-1 Number system **7 (L) hrs**

Introduction to number system - Binary, Octal, Decimal, Hexadecimal. Number base conversions- 1‘s, 2‘s, rth‘s complements, signed Binary numbers. Binary Arithmetic, Binary codes - Weighted BCD, Gray code, Excess 3 code, ASCII – conversion from one code to another.

Unit 2 Boolean Algebra **7 (L) hrs**

Logic gates, NAND-NOR Implementation, Boolean postulates and laws, basic logic functions, standard forms of logic expressions – Sum of Products (SOP), Product of Sums (POS), Minterm, Maxterm. Minimization of Boolean expressions – using Boolean Algebra and Karnaugh map Minimization.

Unit-3 Logic families **4 (L) hrs**

Brief overview of Transistor as a switch, Logic gate characteristics – propagation delay, speed, noise margin, fan-out and power dissipation. Standard TTL and static CMOS gates.

Unit-4 Introduction to VHDL (VHSIC Hardware Description Language) 4 (L) hrs
Behavioural – data flow algorithmic and structural description, lexical elements, data objects types, attributes, operators. VHDL coding examples.

Part-B

Unit-5 Combinational logic 12 (L) hrs
Arithmetic circuits, Magnitude Comparator, decoders, encoders, code converters, parity checker, multiplexers, de-multiplexers, and their use in logic synthesis. Implementation of combinational logic using MUX, Hazards in combinational circuits, combinational circuit design examples in VHDL and simulation.

Unit-6 Sequential Logic Design 12 (L) hrs
Latches and Flip Flops (SR, D, JK, T), Timing in sequential circuits, Shift register, Counters – synchronous, asynchronous, Modulus and Ring Counters. Sequential circuit design examples in VHDL and simulation.

Laboratory Work
To implement the above topics covered with Digital Electronics kits & IC's and VHDL examples for combinational and sequential circuits using XILINX ISE 9.1/9.2.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	To verify the truth table of various logic gates using ICs' 7408, 7402, 7400, 7404, 7432, 7486.
2	To implement basic gates using universal gates NAND and NOR.
3	4-Bit Binary-to-Gray and Gray-to-Binary Code Converter: Realization using XOR gates.
4	4-Bit BCD-to-Excess-3: Realization using basic and universal gates.
5	Multiplexer: Truth-table verification and realization of half adder and full adder using IC74153 chip.
6	Implement Arithmetic Circuits using basic VHDL simulation.
7	Implement 4:1 MUX and 8:1 MUX using VHDL for a given problem.
8	Implement 1:4 DEMUX and 1:8 DEMUX using VHDL for a given problem.
9	Implement SR, JK, D and T Flip Flops using VHDL simulation.
10	Realization of 4-bit counter asynchronous and synchronous counters using VHDL.

Text Books

1. Mano M.M., Ciletti M.D., "Digital Design with an Introduction to the Verilog HDL", 6th Edition, Pearson India, 2006.
2. Thomas L Floyd, "Digital Fundamentals", 11th Edition, Pearson, 2018.

Reference Books

1. A. K. Maini, "Digital Electronics: Principles, Devices and Applications, Wiley, 2007.
2. Michael D. Ciletti, "Advanced Digital Design with the Verilog HDL", 2nd edition, Pearson education, 2017.
3. Roth and Kinney, "Fundamentals of Logic Design", 7th edition, Cengage learning, 2014.
4. Laboratory Manuals.

Online Learning Materials

1. https://www.youtube.com/watch?v=BikBi6ipmyk&list=PLdlPA9pGVVtZF-VSy02wOiUqrPWptjuGt Accessed on April 15, 2025
2. https://www.youtube.com/watch?v=oNh6V91zdPY&list=PLbRMhDVUMnge4gDT0vBWjCb3Lz0HnYKkX Accessed on April 15, 2025
3. https://www.youtube.com/watch?v=CeD2L6KbtVM&list=PL803563859BF7ED8C Accessed on April 15, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Digital System Design	Prof. Neeraj Goel	IIT Ropar	https://onlinecourses.nptel.ac.in/noc21_ee39/preview
2	Digital Electronic Circuits	Prof. Goutam Saha	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc25_ee20/preview
3	Digital Circuits	Prof. Santanu Chattopadhyay	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc21_ee75/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: SMCS101

Course Title: Seminar and Technical Report Writing for Engineers

Programme: B.Tech.	**L:** 0 **T:** 0 **P:** 2	**Credits:** 1
Semester: 3	**Theory/Practical:** Practical	**Teaching Hours:** 30(P) = 30 hrs
Total Max. Marks: 50	**Continuous Assessment (CA) Marks:** 50	**End Semester Examination (ESE) Marks:** Nil
Duration of End Semester Examination (ESE): NA		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Install BibTeX and PDFLaTeX on Windows and Linux operating systems.
2	Utilize LaTeX tools for document preparation and compilation.
3	Compare LaTeX and Markdown formats using Pandoc for documentation purposes.
4	Present technical content effectively through oral presentations.
5	Use LaTeX effectively for technical document writing.

Contents

Experiment No.	Experiment Title
1	Introduction to LaTeX and Its Background • History of LaTeX, its evolution, and use cases. • Comparison with other document preparation tools (Word, Google Docs). • Advantages of LaTeX in research and technical writing.
2	Introduction to Supporting Tools for LaTeX • TeXLive, TeXworks and Overleaf. • Online vs. offline compilation of LaTeX documents.
3	Installation • Installing BibTeX on Windows: step-by-step process. • Installing PDFLaTeX on Linux: dependencies and configurations.
4	Develop a LaTeX script to create a simple document that consists of 2 sections [Section1, Section2], and a paragraph with dummy text in each section. And also include header [title of document] and footer [institute name, page number] in the document.
5	Develop a LaTeX script to create a document that displays the sample Abstract/Summary.

<table>
<tr><td>6</td><td>Develop a LaTeX script to create the Certificate Page of the Report [Use suitable commands to leave the blank spaces for user entry].</td></tr>
<tr><td>7</td><td>Develop a LaTeX script to create a document that contains the following table with proper labels:
<table>
<tr><th rowspan="2">S.No</th><th rowspan="2">URN</th><th rowspan="2">Student Name</th><th colspan="3">Marks</th></tr>
<tr><th>Subject1</th><th>Subject2</th><th>Subject3</th></tr>
<tr><td>1</td><td>4XX22XX001</td><td>Name 1</td><td>89</td><td>60</td><td>90</td></tr>
<tr><td>2</td><td>4XX22XX002</td><td>Name 2</td><td>78</td><td>45</td><td>98</td></tr>
<tr><td>3</td><td>4XX22XX003</td><td>Name 3</td><td>67</td><td>55</td><td>59</td></tr>
</table></td></tr>
<tr><td>8</td><td>Develop a LaTeX script to include the side-by-side graphics/pictures/figures in the document by using the subgraph concept.</td></tr>
<tr><td>9</td><td>Develop a LaTeX script to create a document that consists of two mathematical equations.</td></tr>
<tr><td>10</td><td>Develop a LaTeX script to demonstrate the presentation of Numbered theorems, definitions, corollaries, and lemmas in the document.</td></tr>
<tr><td>11</td><td>Develop a LaTeX script to create a document that consists of two paragraphs with a minimum of 10 citations in it and display the reference in the section.</td></tr>
<tr><td>12</td><td>Develop a LaTeX script to create a simple report and article by using suitable commands and formats of Major/Minor project report.</td></tr>
<tr><td>13</td><td>Markdown and Pandoc
Introduction to .md format and Markdown syntax.
Use cases of Markdown in documentation and technical writing.
Conversion between LaTeX and Markdown using Pandoc.</td></tr>
</table>

Note:

- Students will give 15 minutes presentations on topics related to emerging technologies to showcase their seminar presentation abilities.
- Students will also prepare a simple report based on topic of presentation using formats of Major/Minor project report.
- Evaluation of seminar will be based on Rubrics.

Reference Books

1. Donald Knuth E., "The TexBook", Twentieth revision, Addison Wesley, 1991.
2. Leslie Lamport, "LaTeX: A document preparation system, User's guide and reference manual", 2nd edition, Addison Wesley, 1994.
3. Frank Mittelbach, with Ulrike Fischer "The LaTeX Companion, 3rd edition (TTCT series)", 3rd edition, Addison-Wesley Professional, 2023.
4. Van Emden J., "Effective communication for Science and Technology", Palgrave, 2001.
5. Van Emden J., "A Handbook of Writing for Engineers", Macmillan, 1998.

6. Van Emden J. and Easteal J., "Technical Writing and Speaking, an Introduction", McGraw-Hill, 1996.
7. Pfeiffer W.S., "Pocket Guide to Technical Writing", Prentice Hall, 2000.
8. Eisenberg A., "Effective Technical Communication", McGraw-Hill, 1992.

Online Learning Materials

1. https://latex-tutorial.com/tutorials/ Accessed on April 07, 2025
2. https://guides.nyu.edu/LaTeX/home Accessed on April 07, 2025
3. https://www.overleaf.com/learn Accessed on April 07, 2025
4. https://www.overleaf.com/gallery/tagged/homework Accessed on April 07, 2025
5. https://www.tug.org/twg/mactex/tutorials/ltxprimer-1.0.pdf Accessed on April 07, 2025

Semester 4

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS104
Course Title: Database Management Systems

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 4	**Theory/Practical:** Theory	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply the concepts of database systems, data modeling and relational database to design Entity-Relationship (ER) diagrams and map them to relational schemas.
2	Examine functional dependencies and normalization techniques to improve database schema quality and minimize redundancy.
3	Use SQL commands to define, manipulate, and query relational databases including complex SQL queries and PL/SQL constructs.
4	Understand the data storage structure and transaction management in DBMS.
5	Make use of NoSQL database to store, retrieve, and manipulate data.
6	Design and develop relational databases to address real-world application requirements.

Contents

Part-A

Unit-1 Database Environment and Development Process **4 (L) hrs**

Basic Concepts and Definitions, Traditional File Processing Systems, The Database Approach-Data Models, Relational Databases, Database Management Systems, Advantages of the Database Approach; Components of the Database Environment. The Database Development Process-Systems Development Life Cycle, Three-Schema Architecture for Database Development, Managing the People involved in Database Development.

Unit-2 Relational Model and Database Design using the E-R Model **6 (L) hrs**

Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages- The Relational Algebra. Overview of the Design Process, The Entity-Relationship Model, Complex Attributes, Mapping Cardinalities, Primary Key, Removing Redundant Attributes in Entity

Sets, Reducing E-R diagrams to Relational Schemas, Extended E-R features, Entity-Relationship Design Issues, Alternative Notations for Modeling Data, Other Aspects of Database Design.

Unit-3 Structured Query Languages (SQL) 8 (L) hrs

Overview of the SQL Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Sub-queries, Modification of the Database, Intermediate SQL-Join Expressions, Views, Transactions, Integrity Constraints, SQL Data types and Schemas, Index Definition in SQL, Authorization. Advanced SQL - Accessing SQL from Programming Language, Functions and Procedures.

Part-B

Unit-4 Normalization 6 (L) hrs

The purpose of Normalization, Data redundancy and update anomalies, Functional Dependencies- Characteristics of Functional Dependencies, Identifying Functional Dependencies, Identifying the Primary Key for a Relation. The Process of Normalization- First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF). Advanced Normalization- More on Functional Dependencies - Inference Rules for Functional Dependencies, Minimal Sets of Functional Dependencies. Boyce–Codd Normal Form (BCNF) - Definition of BCNF. Fourth Normal Form (4NF) - Multi-Valued Dependency, Definition of Fourth Normal Form. Fifth Normal Form (5NF)- Lossless-Join Dependency, Definition of Fifth Normal Form

Unit-5 Data Storage Structures 3 (L) hrs

Database Storage Architecture, File Organization, Organization of Records in Files, Data-Dictionary Storage, Database Buffer, Column-Oriented Storage, Storage Organization in Main Memory Databases. Indexing-Basic Concepts, Ordered Indices, B^+ Tree Index Files, Hash Indices.

Unit-6 Transaction Management 10 (L) hrs

Transactions-Transaction Concept, A Simple Transaction Model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability. Concurrency Control-Lock-Based Protocols, Deadlock handling, Multiple Granularity, Insert Operations, Read operations and Predicate Reads, Timestamp-Based Protocols, Validation-Based Protocols. Recovery System-Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm-Log Based Recovery and Shadow Paging.

Unit-7 Advanced Databases 8 (L) hrs

NoSQL Databases-Introduction to NoSQL Systems, The CAP Theorem, Document-Based NoSQL Systems and MongoDB, NoSQL Key-Value Stores, Column-Based or Wide Column NoSQL Systems, NoSQL Graph Databases and Neo4j. Other Advanced Databases- Introduction of Cloud Databases, Spatial and Geographic information system (GIS) Databases and their applications.

Laboratory Work:

Students will create ER Diagrams using any tool, perform operations of relational algebra in RelaX

tool, perform SQL commands to demonstrate the usage of DDL, DML, joining of tables, grouping of data, PL/SQL constructs. They will also implement CRUD operations of NOSQL Databases.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Design ER diagrams for specific problems using any open source tool.
2.	Write queries for Relational Algebra in RelaX tool.
3.	Write SQL queries for Data Definition Languages for creating and managing tables (create, drop, truncate, alter and rename), DEFAULT option, referencing another user's table.
4.	Write SQL queries for Data Manipulation Languages for manipulating data in tables (select, insert, update and delete commands).
5.	Write SQL queries to implement Basic SQL select statements using arithmetic operators, logical operators, comparison operators, operators' precedence, eliminating duplicate rows, using column aliases.
6.	Write SQL queries for Character functions - case manipulation and character manipulation functions; number functions, date functions; using arithmetic operators with dates; date functions, conversion functions.
7.	Write SQL queries for extracting data from more than one table (Cartesian product, equi-Join, non-equi-join, outer join)
8.	Write SQL queries for aggregating, sorting data, using various clauses (group by, having and order by).
9.	Write SQL queries for subqueries, nested queries (single-row subqueries; multiple-row subqueries; using group function in a subquery; HAVING clause with subqueries; usage of operators in multiple-row subqueries)
10.	Write SQL queries to create views and also apply different operations on views.
11.	Write programs to access SQL from programming language using basic PL/SQL constructs.
12.	Write PL/SQL programs to illustrate the concept of functions and procedures.
13.	Apply Normalization Techniques up to third Normal Form(3NF) to a Given Relation: i. Identify the repeating groups and remove them to convert the table into First Normal Form (1NF). ii. Identify partial dependencies and convert the table into Second Normal Form (2NF). iii. Identify transitive dependencies and convert the table into Third Normal Form (3NF). iv. Implement the normalized tables using SQL CREATE TABLE statements.

	v. Insert sample data into normalized tables using SQL INSERT. vi. Demonstrate data retrieval using SQL SELECT with JOIN.
13.	NoSQL Databases-Installation of MongoDB, CRUD operations of NoSQL Databases with Python/Java.
14.	Mini Project By using standard database design rules, database has to be designed for a specific assigned problem to a group of two to three students. ER diagram related to project with an open source database tools like MySQL workbench, Lucidchart must also be prepared. The group of students must submit a project report of 8 to 10 pages (approximately) and the team will have to demonstrate as well as have to give a presentation of the same.

Text Books

1. Abraham Silberschatz, S. Sudarshan, Henry F. Korth, "Database System Concepts", 7th Edition, Tata McGraw - Hill Education, 2020.
2. Shamkant B. Navathe, Ramez Elmasri, "Fundamentals of Database Systems", 7th Edition, Pearson, 2016.
3. Jeffrey A. Hoffer, V. Ramesh and Heikki Topi, "Modern Database Management", 12th Edition, Pearson, 2016.

Reference Books

1. Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", 4th Edition, BPB Publications, 2009.
2. Sadalage, P. & Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", 1st Edition, Wiley Publications 2019.

Online Learning Material

1. https://scs.dypvp.edu.in/documents/e-books/DMBS/database-management-systems-raghu-ramakrishnan.pdf Accessed on May 13, 2025
2. http://infolab.stanford.edu/~ullman/dscb.html Accessed on May 13, 2025
3. https://industri.fatek.unpatti.ac.id/wp-content/uploads/2019/03/162-Introduction-to-Database-Management-System-Satinder-Bal-Gupta-Aditya-Mittal-Edisi-2-2017.pdf Accessed on May 13, 2025
4. https://circuitmix.com/free-download-fundamentals-of-database-management-systems-ebook/ Accessed on May 13, 2025
5. https://freevideolectures.com/course/2668/database-management-system Accessed on May 13, 2025
6. https://www.studytonight.com/dbms/er-to-relational-model.php Accessed on May 13, 2025
7. https://www.studytonight.com/dbms/database-normalization.php Accessed on May 13, 2025

Supplementary SWAYAM Courses

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Database Management System	Prof. Partha Pratim Das, Prof. Samiran Chattopadhyay	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc21_ee 39/preview
2	Database Management System	Prof. Partha Pratim Das	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc19_cs46/preview
3	Introduction to Database Systems	Prof. Sreenivasa Kumar	IIT Madras	https://onlinecourses.nptel.ac.in/noc25_cs40/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS105
Course Title: Operating System

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 4	**Theory/Practical:** Theory	**Teaching Hours:** 45 (L) + 30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand the components of operating systems and design issues associated with it.
2	Apply the knowledge of various process management concepts for process synchronization.
3	Illustrate the concept of scheduling algorithms to work in multitasking environment.
4	Analyze the algorithms for deadlock management, file management and memory management.
5	Evaluate file system structures and implement file handling, directory structures, and disk scheduling algorithms.

Contents

Part-A

Unit-1 Introduction **6(L) hrs**

Concept of Operating Systems, Generations of Operating systems, Types and structures of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, System Programs, System boot, Basics of OS security, user authentication.

Unit-2 Process management and scheduling algorithms **10(L) hrs**

Concept of processes and threads, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. Thread - Definition, Various states and types, Benefits of threads, Multithreading Models. Process Scheduling - Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria, Scheduling algorithms, Multiprocessor scheduling.

Unit-3 Process Synchronization **5(L) hrs**

Inter-process Communication-Shared memory, message passing. Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution- Test and Set instruction, Software solution- Strict Alternation, Peterson's Solution. The Producer/Consumer Problem, Semaphore.

Case study 1: **2(L) hrs**

Implementation of Unix based operating systems and windows based operating Systems.

Part-B

Unit-4 Deadlocks **4(L) hrs**

Introduction to deadlocks, Conditions for deadlock, Resource allocation graphs, Deadlock prevention and avoidance, Deadlock detection and recovery.

Unit-5 Memory Management **10(L) hrs**

Basic concept, Logical and Physical address map, Swapping, Memory allocation - Contiguous Memory allocation, Fixed and variable partition, Internal and External fragmentation and Compaction, overlay, Paging - Structure of the Page Table, Allocation of Frames, Segmentation. Virtual Memory Management - Demand Paging, Page Replacement algorithms, Thrashing.

Unit-6 File and storage Management **6(L) hrs**

Concept of File, Access methods, File types, File operation, Directory structure, Allocation methods (contiguous, linked, and indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table). Secondary Storage: Disk structure, Disk scheduling algorithms.

Case study 2: **2(L) hrs**

Implementation of Mobile Operating systems and Robotics Operating systems.

Laboratory Work: Install and practice Linux/Unix commands, basic shell scripting. Simulate CPU scheduling, Disk Scheduling, Process synchronization, deadlock, Memory Management techniques, page replacement policies, file management techniques using either C or C++.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents

Experiment No.	Experiment Title
1	**Operating System Installation and Virtualization:** a) Installation of Linux operating systems.

	b) Setting up and managing virtual operating systems using Virtual Box.
2	**System Architecture and Types of OS:** a) Study of single-user, multi-user, network operating systems. b) Exploring system boot processes and OS initialization
3	**Linux System Administration:** a) Study of Linux system architecture, booting process, and kernel. b) Exploring system services, logs, and process management.
4	**Linux Commands and Scripting:** a) Executing essential Linux commands for file handling, disk management, and networking. b) Working with permissions, process control, and system monitoring.
5	**Shell Scripting & Automation:** a) Basics of shell scripting (Bash, Zsh). b) Writing scripts with conditional statements, loops, and functions. c) Automating system tasks such as backups, log management, and user creation.
6	**CPU Scheduling Algorithms:** Implement and analyze First Come First Serve (FCFS), Shortest Job First (SJF), Round Robin (RR), and Priority Scheduling and also Compute turnaround time and waiting time.
7	**Process Synchronization & Deadlocks:** a) Simulation of Producer-Consumer problem using semaphores. b) Implementation of Banker's Algorithm for deadlock avoidance.
8	**Memory Management Techniques:** Implement and compare Worst-Fit, Best-Fit, First-Fit. Also, Simulate paging and segmentation techniques.
9	**Page Replacement Policies:** Implementation of FIFO (First In First Out), LRU (Least Recently Used), Optimal Page Replacement.
10	**Disk Scheduling Algorithms:** - Simulation of FCFS (First Come First Serve), SCAN, C-SCAN
11	**File Management Techniques:** - Simulation of file allocation strategies (Sequential, Indexed, Linked)

Text Books

1. A Silberschatz and Peter B. Galvin, "Operating System Concepts", 10th edition, Addison Wesley, 2019.
2. GaryNutt, "Operating Systems Concepts", 3rd edition, Pearson Education Inc. 2004.

Reference Books

1. Dhamdhere, "Systems Programming & Operating Systems", 2nd edition, Tata McGraw Hill, 1996.

2. Tanenbaum A. S, "Operating System Design & Implementation", 3rd edition, Pearson Education Inc. 2006.
3. Charles Crowley, "Operating System: A Design-oriented Approach", Indian edition, Tata McGraw Hill, 2006.

Online Learning Materials

1. http://dinus.ac.id/repository/docs/ajar/Operating_System.pdf Accessed on May 13, 2025
2. https://youtu.be/mXw9ruZaxzQ Accessed on May 13, 2025
3. https://www.coursera.org/learn/os-pku Accessed on May 13, 2025
4. https://www.coursera.org/learn/operating-system-foundations Accessed on May 13, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Introduction to Operating System	Prof. Chester Rebeiro	IIT Madras	https://nptel.ac.in/courses/106106144

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS106
Course Title: Computer Networks

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 2	**Credits:** 5
Semester: 4	**Theory/Practical:** Theory	**Teaching Hours:** 45(L) + 15(T) + 30(P) = 90 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 40%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any) - NIL

Additional Material Allowed in ESE- Scientific Calculator

On completion of the course, the student will have the ability to-

CO#	Course Outcomes
1	Understand the key concepts of data communication to real-world networking scenarios.
2	Apply the concepts of analog and digital signals for transmission of data using efficient bandwidth utilization.
3	Evaluate network performance metrics and algorithms for selecting efficient communication paths.
4	Analyze the design concepts for process to process delivery in end-to-end communication.
5	Assess the functionalities of various network layers and protocols to understand their contribution in computer networks

Contents

Part-A

Unit-1 Introduction **8(L) hrs**

Data Communication Components, Representation of data and data flow, Networks-Network criteria, Physical Structure- Type of connections. Network Type - LAN and WAN. Internet brief History, Protocols- Syntax, Semantics, Timing, Standards and Administration. Networks Models- Protocol Layering, Principle of Protocol Layering, Addressing- Physical Address, Logical Address, Port Address, Specific Address. TCP/IP Protocol suite OSI Model.

Unit-2 Physical Layer **8(L) hrs**

Analog and digital signals, Analog and digital transmission, Data Rate limits- Nyquist formula, Shannon formula. Transmission impairments- Attenuation, Noise, Distortion. Network Performance -Bandwidth, Bit Rate, Throughput, Latency, Propagation Time, Propagation speed. Techniques for

Bandwidth utilization- Frequency division, Time division and Wave division Multiplexing. Concepts of Spread Spectrum. Transmission Media- Guided media- Twisted Pair, Co-axial, Fiber optics with propagation modes, Connectors and its applications. Unguided Media- Radio Wave, Micro Wave, Infra-Red, Ranges, Propagation Methods and its Applications. Switching- Circuit, Message and Packet switching and their comparison.

Unit-3 Data Link Layer **8(L) hrs**

Types of Errors, Detection- Redundancy, Parity, Cyclic Redundancy Check and Checksum. Error Correction- Forward Error Correction- Hamming Code, Burst Error correction. Data Link Control and Protocols- Error control, Flow control. Flow and error control Mechanisms: Stop and Wait ARQ, Go-Back-N ARQ and Selective Repeat ARQ, Piggybacking. Multiple Access Random Access- Concept of Pure ALOHA and Slotted ALOHA. CSMA- Persistent methods, CSMA/CD, CSMA/CA.

Part-B

Unit-4 Network Layer **11(L) hrs**

Logical addressing- Classful and classless, IPv4 and IPv6 Address Format. Network Layer Protocols- ARP, RARP - Concept and Format of ARP Packet. DHCP- Message Format and working operation. IPv4 and IPv6- Header Format and Comparison. Routing algorithms- Concept of Routing Table, Static and Dynamic routing. Unicast Routing- Inter-domain and Intra-domain. Intra-domain- Distance vector routing, Link State Routing- Dijkstra Algorithm. Inter-domain- Path Vector Routing. Congestion Control Algorithms- Leaky Bucket and Token Bucket Algorithms. Congestion Control Polices- Open-loop Congestion Control, Closed-loop Congestion Control.

Unit-5 Transport Layer **6(L) hrs**

Services- Process to Process Communication, Addressing, Connectionless and Connection Oriented Services, Multiplexing and De-Multiplexing, Flow Control, Error Control. User Datagram Protocol (UDP)- Header Format and its Applications. Transmission Control Protocol (TCP)- Header Format and its Applications. Elements of Transport Protocols- Three way handshaking Connection Establishment and Release.

Unit-6 Session, Presentation and Application Layer **4(L) hrs**

Data Compression Techniques- Lossless and Lossy Compression Techniques. Domain Name Space (DNS)- Purpose, Namespace, DNS in the Internet, Resolution. TELNET- Local versus Remote Logging, EMAIL- Architecture, FTP (File Transfer Protocol)- Control Connection, Data Connection, FTPS, WWW (World Wide Web)- Architecture HTTP (Hyper Text Transfer Protocol), HTTPS.

Laboratory Work: *Students will simulate the scenario related to above topics with any open-source simulation tools and implement programs with any open source C++ tools.*

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents

Experiment No.	Experiment Title
1	Familiarization with various network simulation tools: Wireshark, CISCO packet tracer, NS2/NS3, GNS3 etc.
2	Simulation of message transmission for single point and multi point devices.
3	Study and implementation of various LAN topologies any simulation tools.
4	Configuration of TCP/IP Protocols in Windows.
5	Familiarization with various transmission media and its specifications.
6	Preparing straight and cross cables using crimping tool.
7	Familiarization with networking components LAN Adapters, Hubs, Switches, Routers etc.
8	Develop a program to compute the Hamming Distance between any two code words (Decimal to binary conversion first)
9	To implement and check the error detection/error correction techniques in networks using a C++ program using Single bit parity.
10	To implement and check the error detection/error correction techniques in networks using a C++ program using even odd parity.
11	To implement and check the error detection/error correction techniques in networks using a C++ program using CRC.
12	Designing and implementing class A, B and C networks
13	To study and use of the basic networking commands.: arp –a, hostname, ipconfig, ipconfig /all, Ipconfig /renew, Ipconifg /release, ipconfig /flushdns, nbtstat –a, netdiag, netstat, nslookup, pathping, ping, route and tracert in Windows.
14	Configuration of WWW protocols using any simulation tool.

Text Books

1. Behrouz A. Forouzan, “Data Communication & Networking”, Tata McGraw Hill, 5th edition, 2017
2. Andrew S. Tanenbaum, “Computer Networks”, Pearson Education, 6th edition, 2022
3. Douglas E. Comer, “Internetworking with TCP/IP”, Volume-I, Prentice Hall, India. 6th edition, 2015

Reference Books

1. James F. Kurose and Keith W. Ross, “Computer Networking”, Pearson Education. 7th edition,2016
2. Behrouz A. Forouzan, “Data Communication & Networking”, Tata McGraw Hill, 3rd edition, 2004
3. Behrouz A. Forouzan, “Data Communication & Networking”, Tata McGraw Hill, 4th edition, 2007
4. W. Stallings, “Data and Computer Communication”, Prentice Hall of India. 10th edition, 2017

Online Learning Materials

1. https://ocw.mit.edu/courses/6-263j-data-communication-networks-fall-2002/resources/lecture1/
Accessed on May 15, 2025

2. https://ocw.mit.edu/courses/6-263j-data-communication-networks-fall-2002/resources/lecture2/ Accessed on May 15, 2025
3. https://ocw.mit.edu/courses/6-263j-data-communication-networks-fall-2002/pages/lecture-notes/ Accessed on May 15, 2025
4. https://youtu.be/lnU-Zw3NEEQ?si=r5ZRdgUkFtN9Vo2V Accessed on May 15, 2025
5. https://youtu.be/29Qdz0FmvmQ?si=dtsXLXJfDpGOfHFY Accessed on May 15, 2025
6. https://youtu.be/8BK70UDgyrc?si=vJGSwp74De124bNc Accessed on May 15, 2025
7. https://www.youtube.com/watch?v=9ycQolopz6g Accessed on May 15, 2025
8. https://youtu.be/5AHp0f0489E?si=ZBrq84ewy3aVy-Qy Accessed on May 15, 2025
9. https://youtu.be/8-3CSAkscYU?si=mcLKFV2fZu_MKQdQ Accessed on May 15, 2025
10. https://youtu.be/5ex1s4lURto?si=28ULjxNjzz7uHJUD Accessed on May 15, 2025
11. https://youtu.be/JsUzPkOxZfA?si=Tx50yS7Hab86iu2p Accessed on May 15, 2025
12. https://youtu.be/qZlMS4yJM-E?si=WOHubr2UDL7htzO9 Accessed on May 15, 2025
13. https://youtu.be/VUdfS70puWI?si=qjvKdGrDblRJADNk Accessed on May 15, 2025
14. https://youtu.be/f1y25BfOH9I?si=jePNPqn665KMpBgm Accessed on May 15, 2025
15. https://youtu.be/5vbPS-KnhvI?si=Fgn7eb1r7Cip7Riv Accessed on May 15, 2025
16. https://youtu.be/rGWPOm3ectk?si=grDN6_KyOxuyNjrW Accessed on May 15, 2025
17. https://youtu.be/i6wlR8dmw1k?si=vvGG8bl5H_yFASYW Accessed on May 15, 2025
18. https://youtu.be/JkqgkN_j6Lo?si=T9CHaEeGAl99Gv6J Accessed on May 15, 2025
19. https://youtu.be/8LeqAH_ppsA?si=e4nmBiddE_EEk1cu Accessed on May 15, 2025
20. https://www.youtube.com/watch?v=Lo_pIjzej1A&list=PLfmqK5mMBWj_J-O4jMyYSz4WMZjUQX0Iq Accessed on May 15, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Emergence of Networks & Reference Models (English)	Prof. Sujoy Ghosh	IIT Kharagpur	https://nptel.ac.in/courses/106105081
2	Computer Networks and Internet Protocols (English)	Prof. Soumya Kanti Ghosh and Prof Sandip Chakraborty	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc25_cs15/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS107
Course Title: Discrete Mathematics

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 0	**Credits:** 4
Semester: 4	**Theory/Practical:** Theory	**Teaching Hours:** 45(L) + 15(T) = 60 hrs
Total Max. Marks: 100	**Continuous Assessment (CA) Marks:** 40	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 90%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

Additional Material Allowed in ESE: Scientific Calculator

On completion of the course, the student will have the ability to

CO#	Course Outcomes
1	Apply the principles of sets, relations, and functions to solve mathematical problems.
2	Construct formal proofs to validate logical arguments using mathematical reasoning.
3	Evaluate combinatorial problems using appropriate mathematical methods and techniques.
4	Apply number theory concepts to solve computational problems in computer science.
5	Prove elementary properties of algebraic structures and apply them in the analysis and interpretation of data to draw valid conclusions.
6	Make use of appropriate mathematical models to assess real-world problems.

Contents

Part-A

Unit-1 Fundamentals of Sets, Relations and Functions 9 (L) hrs

Sets – Operations on sets, Subsets, Types of sets, ordered pairs, Proofs of general identities of sets, Classes of sets and partitions, Countable and uncountable sets. Relations –Representations of relations, Types of relations, Composition of relations, Closure properties of relations, Equivalence relations, Compatibility relations, Partial order relations. Functions – Introduction and types of functions, Sum and product of functions, Hashing functions, recursively defined functions.

Unit-2 Prepositional and Predicate Logic 4(L) hrs

Prepositions and compound prepositions, Logical connectives, Truth tables, Logical implication and logical equivalence, Normal forms– Conjunctive and Disjunctive, Validity of well-formed formula, Propositional inference rules–Modus ponens and modus tollens. Predicate logic, Universal and existential quantification, Limitations of propositional and predicate logic.

Unit-3 Combinatorial Mathematics **9 (L) hrs**

Basic counting principles, Permutations and combinations, Pigeonhole principle, Inclusion and exclusion principle, Recurrence relations – Solving homogeneous and non- homogeneous recurrence relations, Sequences, Generating function

Part- B

Unit-4 Number Theory **7 (L) hrs**

The division algorithm, modular arithmetic, Primes and greatest common divisors: The fundamental theorem of arithmetic, Euclidean algorithm. Congruences: Congruence relation, Linear congruence equation, The Chinese remainder theorem, Applications of Congruences.

Unit-5 Algebraic Structures and Morphism **8 (L) hrs**

Algebraic structures with one binary operation, Properties of an operation, Congruence relation, Semi groups, Monoids, Groups, Substructures, Cyclic groups, Cosets, Normal subgroups, Dihedral groups, Permutation Groups. Homomorphism and isomorphism of groups, Applications of groups. Algebraic structures with two binary operations, Rings – Introduction, Abelian ring, Ring with unity, Multiplicative inverse, Subrings, Homomorphism of rings. Integral Domain, Ideals.

Unit-6 Graph Theory **8 (L) hrs**

Graphs – Definition, degree, Connectivity, path, cycle, Directed and undirected, Sub Graph, Bi-connected component and Articulation points. Eulerian chains and cycles, Hamiltonian chains and cycles, shortest paths algorithms – Dijkstra's algorithm, Warshall's algorithm. Rooted trees, Spanning tree algorithms – Kruskal's algorithm, Prim's algorithm. Graph coloring, Map Coloring, Chromatic number, Planar graphs, Euler's formula, Isomorphism and homomorphism of graphs, Applications of graph theory.

Text Books

1. K.H. Rosen, "Discrete Mathematics and its applications", 8th edition, Tata McGraw Hill, 2021.
2. S. Lipschutz and M.Lipson, "Schaum's Outline of Discrete Mathematics", 4th edition, Tata McGraw Hill, 2022.
3. A. Doerr and K. Levarseur, "Applied Discrete Structures for Computer Science", 3rd edition, Pearson Education, Inc. 2018.
4. Tremblay, J.P. and Manohar R., "Discrete Mathematical Structures with Applications to Computer Science", 1st edition, Tata McGraw Hill, 2008.

Reference Books

1. C.L. Liu and D. Mohapatra "Elements of Discrete Mathematics: A Computer Oriented Approach", 4th edition, Tata McGraw Hill, 2014.
2. Susanna S. Epp, Discrete Mathematics with Applications, 5th edition, Wadsworth Publishing Co. Inc. 2020.
3. Haggard G., Schlipf J. and Whitesides, Sue, Discrete Mathematics for Computer Science, 2nd edition, Cengage Learning, 2008.
4. Johnsonbaugh R., Discrete Mathematics, 7th edition, Pearson Education, 2007.

Online Learning Materials

1. Discrete Mathematics and its applications by Kenneth H. Rosen https://mrce.in/ebooks/Maths-Discrete Mathematics & its Applications 8th Ed.pdf
   Accessed on April 30, 2025
2. Discrete Mathematics: An Open Introduction by Oscar Levin
   https://discrete.openmathbooks.org/pdfs/dmoi-tablet.pdf
   https://teachmint.storage.googleapis.com/public/224619601/StudyMaterial/35957954-ed0c-4c03-9772-f1b2a3dfc756.pdf
   Accessed on April 30, 2025
3. Applied Discrete Structures by Al Doerr & Ken Levasseur
   https://math.libretexts.org/Bookshelves/Combinatorics_and_Discrete_Mathematics/Applied_Discrete_Structures_(Doerr_and_Levasseur)
   Accessed on April 30, 2025

Online Courses and Video Lectures

1. https://www.youtube.com/watch?v=pMzcLG6s3z0 Accessed on April 30, 2025
2. https://youtube.com/playlist?list=PLU6SqdYcYsfLV24T0XVb3z3mjl8QG0EBN&si=cbnY62VBzqqGUsP3 Accessed on April 12, 2025
3. https://www.youtube.com/watch?v=NW-naslChdo Accessed on April 12, 2025
4. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics- for- computer-science-fall-2010/video-lectures
   Accessed on April 12, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Discrete Mathematics	Prof. Sudarshan Iyengar, Prof. Anil Shukla	IIT Ropar	https://onlinecourses.nptel.ac.in/noc24_cs92/preview
2	Discrete Mathematics	Prof. Sudarshan Iyengar, Prof. Prabuchandran K.J	IIT Ropar, IIT Dharwad	https://onlinecourses.nptel.ac.in/noc22_cs33/preview
3	Discrete Mathematical Structures	Prof. Kamala Krithivasan	IIT Madras	https://nptel.ac.in/courses/106106094
4	Discrete Mathematics	Dr. Sugata Gangopadhyay, Dr. Aditi Gangopadhyay	IIT Roorkee	https://nptel.ac.in/courses/111107058

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS108
Course Title: Artificial Intelligence

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 4	**Theory/Practical:** Theory	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 60%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): ESC105

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand the foundational principles and scope of Artificial Intelligence and its role in building intelligent systems.
2	Examine problems that can be solved using AI techniques such as search, logic, learning, and decision-making.
3	Apply AI methods and models, including classical algorithms, machine learning, and neural networks to solve real-world problems.
4	Demonstrate the ability to use modern AI tools and platforms to develop intelligent systems and evaluate their performance.
5	Analyze the capabilities and limitations of AI technologies, including deep learning, generative models, and large language models.
6	Exhibit awareness of ethical, societal, and practical considerations in the design and deployment of AI systems.

Contents

Part-A

Unit-1 Introduction **3 (L) hrs**

Intelligence, Foundations of Artificial Intelligence (AI), History of AI; Agents and Environments - The concept of rationality, Nature of environments, Structure of Agents, Application and Impact of AI in real world.

Unit-2 Problem Formulation and Solution **10 (L) hrs**

Problem-Solving agents, Search Algorithms-Uninformed Search Strategies – Breadth first search, Depth first search, Iterative Deepening, Informed Search Strategies; Heuristic Search – Designing a heuristic function, Branch and Bound, Hill Climbing, Best-first Search, A* algorithm, Iterative deepening A*(IDA), Small memory A*(SMA).

Unit-3 Game Playing **5 (L) hrs**

Representation of games as search problems, Understanding game trees, Perfect Information game, Imperfect Information game, Evaluation function, Minimax algorithm, Alpha-beta pruning.

Unit-4 Logical Reasoning **5 (L) hrs**

Propositions, Inference in Propositional logic and First order Predicate logic, Resolution, Forward chaining, Backward chaining, Knowledge representation techniques - Semantic networks, Frames.

Part-B

Unit-5 Planning **7 (L) hrs**

Basic representation of plans, Partial order planning, Planning in the blocks world, Hierarchical planning, Conditional planning, Representation of time, schedule and resource constraints, Measures, temporal constraints.

Unit-6 Uncertainty **5 (L) hrs**

Basic probability, Bayes rule and its use, Belief networks, Default reasoning, Fuzzy sets and fuzzy logic; Decision making– Utility theory, Utility functions, Decision theoretic expert systems.

Unit-7 Introduction to Machine Learning and Deep Learning **5 (L) hrs**

Introduction to machine learning, Types of learning - Supervised, Unsupervised, Reinforcement Learning, Introduction to neural networks and deep learning.

Unit-8 Advancements in AI **5 (L) hrs**

Explainable AI (XAI), Edge AI, Generative AI, AI for Social Good - Applications of AI in healthcare, education, climate change, and sustainability, AI Security - Protecting AI systems from attacks and ensuring secure AI development), Case studies of AI-driven innovations - Smart Speakers, Self-Driven cars, Smart Agriculture, etc.

Laboratory Work
Students will implement above topics in python or similar programming language and will also implement mini project based on course contents.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents

Experiment No.	Experiment Title
1	Implement a simple reflex agent for the vacuum-cleaner world.

2	Implement Breadth First and Depth First search for water jug problem.
3	Implement Hill Climbing search algorithm.
4	Implement A* algorithm for eight puzzle problem.
5	Develop a Tic Tac Toe game using the Minimax algorithm with alpha-beta pruning.
6	Represent a small real-world domain using frames and semantic networks.
7	Implement the Goal Stack Planning method for the Blocks World.
8	Implement Bayes Rule to classify emails as spam or not spam.
9	Build a small neural network from scratch without using any deep learning libraries.
10	Generate images from text using a pre-trained diffusion or GAN model.

Mini Project: Student will develop a project assigned from course contents in a group of students. They must submit a project report and give a presentation of the same.

Text Books

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, Fourth Edition, 2022.
2. Saroj Kaushik, "Artificial Intelligence", Cengage Learning India, Second Edition, 2022.

Reference Books

1. David L. Poole, Alan K. Mackworth, "Artificial Intelligence", Cambridge University Press, Second edition, 2017.
2. Elaine Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw Hill, 3rd edition, 2019.
3. Trivedi, M.C., "A Classical Approach to Artificial Intelligence", Khanna Publishing House, Delhi.
4. A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd edition, Sebastopol, CA: O'Reilly Media, 2022.

Online Learning Materials

1. https://www.youtube.com/watch?v=V0fIteSbu8M Accessed on August 4, 2025
2. Search - Lecture 0 - CS50's Introduction to Artificial Intelligence with Python 2020 Accessed on August 4, 2025
3. An Introduction to Artificial Intelligence | Prof. Mausam Accessed on August 4, 2025
4. OpenAI, HuggingFace documentation
5. https://platform.openai.com Accessed on August 4, 2025
6. https://teachablemachine.withgoogle.com Accessed on August 4, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Artificial Intelligence: Search Methods for Problem Solving	Prof. Deepak Khemani	IIT Madras	https://onlinecourses.nptel.ac.in/noc25_cs88/preview
2	Artificial Intelligence: Concepts and Techniques	Prof. V. Susheela Devi	IISc Bangalore	https://onlinecourses.nptel.ac.in/noc25_cs159/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: LCCS109
Course Title: Data Analytics Tools

Programme: B.Tech.	**L:** 0 **T:** 0 **P:** 2	**Credits:** 1
Semester: 4	**Theory/Practical:** Practical	**Teaching Hours:** 30(P) = 30 hrs
Total Max. Marks: 50	**Continuous Assessment (CA) Marks:** 50	**End Semester Examination (ESE) Marks:** Nil
Duration of End Semester Examination (ESE): NA		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply data preparation techniques in Excel and Python to clean, filter, and merge structured datasets for analysis.
2	Create charts and dashboards using Matplotlib and Power BI/Tableau to display the changes and relationships within datasets.
3	Perform statistical inference methods to validate hypotheses using real-world data samples.
4	Design interactive dashboards in Power BI/Tableau to present key performance indicators and filtering options for data exploration.
5	Integrate Excel, Python, and data visualization tools in a project to analyze a given dataset by preparing data, calculating performance metrics, and presenting insights for decision-making.

Contents

Experiment No.	Experiment Title
1	Introduction to Data Analytics Tools: Overview of data analytics and its lifecycle, Comparison of tools- Excel, Python (Pandas, NumPy), Power BI, Tableau, Setting up the environment- Jupyter Notebook, Google Colab, VS Code 1.1 Install Python and set up Jupyter Notebook. Load a CSV dataset containing student records using Pandas. Display the structure, summary statistics, and identify null values.
2	Data Handling with Excel: Working with structured data in Excel, Basic functions-sort, filter, pivot tables, lookup functions. 2.1 Using a dataset of student scores, create pivot tables showing average scores per subject. Add slicers to filter by class/section and generate bar charts for top 10 performers. 2.2 Use Excel functions (VLOOKUP, IF, AVERAGEIF) to identify students with less than 75% attendance and flag them for academic warning

3	Data Manipulation using Python (NumPy & Pandas): Introduction to Pandas DataFrames and NumPy arrays, Data cleaning: missing values, duplicates, outliers, Data transformation: filtering, grouping, merging, reshaping. 3.1 Clean a merged dataset containing student marks and attendance. Handle missing values, drop duplicates, and create a new column final_score = 0.7 * theory + 0.3 * practical 3.2 Perform data transformations: • Group by subject and compute average scores. • Use pivot_table to view subject-wise performance. • Merge two data sources (scores and demographics)
4	Data Visualization Technique: Importance of visualization in analytics, Plotting libraries- Matplotlib, Choosing appropriate charts for different data types. 4.1 Create the following using Matplotlib: • Histogram of student scores • Box plot comparing marks across subjects • Line plot of attendance trends over time
5	Power BI / Tableau Basics: Data import and model creation, Visualization creation and dashboard design, Basic DAX formulas (for Power BI). 5.1 Import student data into Power BI. Create relationships between student, attendance, and marks tables. Use DAX to calculate: • Average score • Attendance ratio • Count of students scoring above 85% 5.2 Design a dashboard in Power BI/Tableau with: • KPI tiles (avg score, attendance %) • Charts for marks by class • Filters/slicers for gender, section, subject
6	Statistical Inference & Hypothesis Testing: Hypothesis formulation, Level of significance, critical region, critical value, Small-sample tests- Student's t-test (single mean, difference of means), F-test (variance comparison), Chi-square test, Large-sample tests- Z-test for single proportion, Z-test for single mean 6.1 Perform a t-test to determine if the average performance of Class A differs significantly from the expected pass threshold (e.g., 50%). 6.2 Conduct an F-test to compare score variability between two sections and determine if one section is more consistent. 6.3 Apply a Chi-square goodness-of-fit test to see if the grade distribution (A, B, C, D, F) matches the expected distribution. 6.4 Use a Z-test to assess whether the proportion of students attending more than 80% of classes is significantly greater than 60%

Mini-Project: Academic Performance Analytics for Student Success

You are hired as a data analyst for a school group that wants to evaluate the performance of students across multiple departments. The aim is to identify:

- Factors affecting student performance
- Analysis of attendance behaviour and subject-wise performance variation
- Whether performance is improving year over year
- If there are statistically significant differences between student groups

Dataset Requirements:
You can use a real or simulated dataset that contains: Student_ID, Name, Department, Class, Gender, Marks_Subject1, Marks_Subject2, Attendance, Grade, Academic_Year

Reference Books

1. Martin C. Brown, "Python The complete Reference", 4th edition, McGraw Hill Education, 2018.
2. Allen B. Downey, "Think Python", 1st edition, Green Tea Press, 2012.
3. Allen B. Downey, "Think Stats", 2nd edition, Green Tea Press, 2014.

Online Learning Materials

1. Wes McKinney, "Python for Data Analysis", 1st Edition, O'Reilly Media, GitHub Repository https://github.com/wesm/pydata-book Accessed on May 12, 2025
2. Microsoft, "Power BI Guided Learning", Microsoft Learn, Free Access. https://learn.microsoft.com/en-us/training/powerplatform/power-bi Accessed on May 12, 2025
3. Tableau Software, "Tableau Free eLearning Videos", Tableau Public, Free Access. https://www.tableau.com/learn/training Accessed on May 12, 2025
4. Excel Easy, "Excel Tutorial for Beginners", Excel-Easy.com, Free Access https://www.excel-easy.com/ Accessed on May 12, 2025

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: MCCS101
Course Title: Environmental Sciences and Sustainability

Programme: B.Tech.	**L:** 2 **T:0 P:** 0	**Credits:** 0
Semester: 4	**Theory/Practical:** Theory	**Teaching Hours:** 30(L) = 30 hrs
Total Max. Marks: 50	**Continuous Assessment (CA) Marks:** 50	**End Semester Examination (ESE) Marks:** 0
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 0%		
Duration of End Semester Examination (ESE): Nil		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	To understand problems of environmental pollution, its impact on human and ecosystem.
2	Gain knowledge on natural processes and resources that sustain life and govern economy.
3	Analyze the role of conservation of biodiversity and its importance
4	To make use of green computing fundamentals for sustainable solutions.
5	To analyze the e-waste management for sustainable development.
6	Adopt sustainability as a practice in life and society.

Contents

Part-A

Unit-1 Introduction to Environment Sciences **5(L) hrs**

Multidisciplinary nature of environmental studies- definition, scope and importance, need for public awareness. Natural Resources- renewable and non-renewable resources, natural resources and associated problems, forest resources, water resources, mineral resources, food resources, energy resources, land resources, role of an individual in conservation of natural resources, equitable use of resources for sustainable life style.

Unit-2 Eco system **5(L) hrs**

Ecosystems- concept of an ecosystem, structure and function of an ecosystem, producers, consumers and decomposers, energy flow in the ecosystem, ecological succession, food chains, food webs and ecological pyramids, introduction, types, characteristic features, structure and function of the following ecosystems- forest ecosystem, grassland ecosystem, desert ecosystem, aquatic ecosystems.

Unit-3 Biodiversity and its conservation **5(L) hrs**

Biodiversity and its conservation- introduction, genetic, species and ecosystem diversity, biogeographical classification of India, Value of biodiversity, biodiversity at global, national and local levels, India as a mega-diversity nation, hot-spots of biodiversity, threats to biodiversity, endangered and endemic species of India, conservation of biodiversity- in-situ and ex-situ conservation of biodiversity.

Part-B

Unit-4 Electronic Waste Management **5(L) hrs**

What is e-waste, raw materials present in e-waste, environmental aspects of e-waste, social aspects of e-waste. An overview of treatment technologies of E-waste- Introduction, types of contaminants in E-waste, treatment strategies of E-waste, recycling, landfill disposal, biological treatment, advanced methods, Case study- E-waste Streams.

Unit-5 Sustainability and Sustainable Computing **6(L) hrs**

Concept of sustainability, Sustainability Indicators, environmental indicators (e.g., air/water quality), needs and challenges- economic, social and aspects of sustainability, energy efficient computing, sustainable software design, green data centers, cloud sustainability. Case study- Green IT.

Unit-6 Green Computing **4(L) hrs**

Understanding Green Computing, environmental challenges in software development, practices for green computing in software development, strategies for promoting environmental sustainability in software development, impacts of green technology, green awareness, green initiatives in information technology, green technology certifications, issues & challenges ahead.

Text Books

1. D.D. Mishra, “Fundamental concepts in Environmental Studies”, S Chand and Co Ltd. 2nd Edition, 2022.
2. Anubha Kaushik and C. P. Kaushik’s “Perspectives in Environmental Studies”, 6th Edition, New Age International Publishers ,2018.
3. Vishal Jain, Murali Raman, “Convergence Strategies for Green Computing and Sustainable Development”, Publisher-IGI Global, 2023.

Reference Books

1. Erach Barucha, “Textbook of Environmental studies”, UGC, 3rd edition, 2021.
2. Benny Joseph, “Environmental Science and Engineering”, Tata McGraw-Hill, New Delhi, 2nd edition, 2016.
3. Dharmendra S. Sengar, “Environmental law”, Prentice Hall of India Pvt. Ltd, New Delhi,2nd edition, 2007.
4. P. Jayarama Reddy, “Urban Mining and Sustainable Waste Management”, CRC Press, 2016.

Online Learning Materials

1. https://wjaets.com/sites/default/files/WJAETS-2024-0052.pdf Accessed: May 12, 2025
2. https://wasteaid.org/wp-content/uploads/2022/06/Handbook-of-E-waste-management.pdf Accessed: May 12, 2025
3. https://onlinecourses.nptel.ac.in/noc24_hs160/preview Accessed: May 12, 2025
4. https://www.ugc.gov.in/oldpdf/modelcurriculum/env.pdf Accessed: May 12, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Certificate course in Environmental Sustainability	Dr. M. Rajesh, Dr. Sindhu P Nair, Dr. Jalajakumari VT, Dr. P Unnikrishnan	IGNOU Regional Centre, Vatakara	https://onlinecourses.swayam2.ac.in/nou20_ag12/preview

Semester 5

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS110
Course Title: Theory of Computation

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 0	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Theory	**Teaching Hours:** 45(L)+15(T) = 60 hrs
Total Max. Marks: 100	**Continuous Assessment (CA) Marks:** 40	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Illustrate and analyze the usage of different types of finite automata (DFA, NFA, Moore and Mealy machines), and apply appropriate transformations to solve automata-based problems.
2	Explain the relationships among formal languages, grammar types, and language classes using the Chomsky hierarchy.
3	Apply the concepts of regular grammars, context-free grammars, and finite automata for language recognition and its simplification.
4	Design pushdown automata based on their computational capabilities to recognize and generate context-free languages.
5	Apply the principles of Turing machines to design computational models for solving complex engineering and decision problems.
6	Analyze and compare the computational capabilities of linear bounded automata with Turing machines in the context of context-sensitive languages.

Contents

Part-A

Unit-1 Finite Automata **7(L) hrs**

Definitions of Finite State Machine, Transition systems, Deterministic Finite Automata (DFA), Acceptance by Finite Automata, Non-Deterministic Finite Automata, Equivalence of DFA and NDFA,

Moore and Mealy machines, Equivalence of Moore and Mealy machine, Minimization of Finite Automata, Applications and Limitations of Finite Automata.

Unit-2 Formal Languages **5(L)hrs**

Formal Languages and Regular Grammar: Basics of Alphabets, Strings, Language, Regular Expression, Grammar, Formal language, Chomsky classification of languages, Languages and their relation, Operations on languages, Closure properties of language classes.

Unit-3 Regular Grammar **8(L) hrs**

Regular grammars, Regular expressions, Operators of Regular Expression and their precedence, Algebraic Laws for Regular Expressions, Kleen's Theorem, Regular Expression to Finite Automata, DFA to Regular Expression -Algebraic method using Arden's Theorem, Equivalence of Finite Automata and Regular expressions, Properties of Regular Languages, Pumping Lemma

Part-B

Unit-4 Context Free Language **8(L)hrs**

Derivation Tree, Ambiguity, Simplification of Context Free Grammar, Normal forms– Chomsky Normal Form, Greibach Normal Form, Pumping lemma.

Unit-5 Push Down Automata **8(L)hrs**

Description and Definition, Acceptance by Push Down Automata, Equivalence of Push Down Automata and Context Free Grammars and Languages.

Unit-6 Turing Machine Linear Bounded Automata **9(L) hrs**

Definition and Model, Representation of Turing Machine, Design of Turing Machine, Variants of Turing Machine, Decidability and recursively enumerable languages, Halting problem, Post correspondence problem, Context sensitive language, Model of linear bounded automata, Relation between linear bounded automata and context sensitive language.

Text Books

1. Hopcroft, J. E., Motwani, R., & Ullman, J. D. "Introduction to Automata Theory, Languages, and Computation", 3rd edition, Pearson Education India, 2008.
2. Mishra, K. L. P., & Chandrasekaran, N. "Theory of Computer Science: Automata, Languages and Computation", 3rd edition, PHI Learning Pvt. Ltd. 2011.

Reference Books

1. Sunitha, K. V. N., & Kalyani, 'N. Formal Languages and Automata Theory", 1st edition, Pearson, 2015.
2. Daniel, A.Cohen, "Introduction to Computer Theory", 2nd edition, John Wiley& sons Inc.

3. M. Sipser, "Introduction to the Theory of Computation", 3rd edition, Cengage Learning, 2013.
4. Peter Linz, "An Introduction to Formal Languages and Automata', 6th edition, Jones and Bartlett Publishers, 2017.

E-Books and Online Learning Material

1. Jean-Eric Pin, Mathematical Foundations of Automata Theory, Lecture notes LIAFA, Université Paris https://www.irif.fr/~jep//PDF/MPRI/MPRI.pdf
2. Michael Sipser, Introduction to the Theory of Computation, Thomson Course Technology http://en.bookfi.net/book/1139836
3. Anil Maheshwari and Michiel Smid, Introduction to Theory of Computation, School of Computer Science, Carleton University, Ottawa Canada https://cglab.ca/~michiel/TheoryOfComputation/TheoryOfComputation.pdf
4. Michael Levet, Theory of Computation‖, Lecture Notes, University of South Carolina-Columbia http://people.math.sc.edu/mlevet/Lecture_Notes.pdf

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1.	Theory of Computation	Prof. Raghunath Tewari	IIT Kanpur	https://nptel.ac.in/courses/106104148
2.	Theory of Computation	Prof. Somenath Biswas	IIT Kanpur	https://nptel.ac.in/courses/106104028
3.	Theory of Automata, Formal Languages and Computation	Prof. Kamala Krithivasan	IIT Madras	https://nptel.ac.in/courses/106106049

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS111

Course Title: Cyber Security

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): CCS106

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply cyber security principles within legal, regulatory, and ethical frameworks of cyberspace.
2	Analyze cyber security threats and attacks impacting systems, networks, and applications.
3	Explore cyber security strategies, risk management practices, and emerging approaches to strengthen organizational security.
4	Examine security protocols and defence mechanisms for secure communication and protection.
5	Evaluate ethical, social, and human factors influencing cyber security practices and incidents.
6	Design preventive and response measures based on real-world cyber security incidents.

Contents

Part A

Unit 1 Introduction to Cyber Security **7 (L)hrs**

Definition and scope of cyber security and cybercrime, classification of cybercrimes, relationship between cybercrime and cyber security, legal and regulatory frameworks for cyber security, IT Act (2000), IT Act Amendments (2008) and key sections, global perspective on cybercrime and cyber security, cyber security awareness, and strategies for safe and responsible use of cyberspace.

Unit 2 Cyber Security Threats and Attacks **7 (L)hrs**

Overview of Cyber Security Threats and Attacks, Proxy servers and anonymizers, phishing attacks, password cracking techniques, keyloggers and spyware, viruses and worms, Trojan horses and

backdoors, steganography, denial of service and distributed denial of service attacks, SQL injection attacks, buffer overflow attacks, ransomware and cryptojacking, exploit kits and zero-day attacks.

Unit 3 Cyber Security Strategies and Emerging Technologies 7 (L)hrs

Security governance and frameworks (ISO 27001, NIST), risk management and assessment, security policies and standards, business continuity and disaster recovery planning, security auditing and compliance, cloud computing security challenges, mobile and wireless device security, BYOD, Fundamentals of Defence-in-Depth Security, Basics of Zero Trust Security Model, security-by-design, DevSecOps Principles, Quantum security and Post-Quantum security Awareness.

Part B

Unit 4 Protocols and Techniques in Cyber Security 11 (L)hrs

Fundamentals of security protocols and secure communication, conceptual study of SSL/TLS for protecting data in transit, functional aspects of IPsec in network-layer security, working principles of SSH for secure remote system access, role of HTTPS in securing web-based communication, fundamental mechanism of Kerberos authentication, essential features of secure email systems (PGP and S/MIME), core ideas behind wireless security standards (WPA, WPA2, WPA3), common challenges and limitations of widely used security protocols, network security techniques (firewalls, IDS/IPS), malware detection and prevention techniques, vulnerability assessment and penetration testing, incident response and recovery techniques, security monitoring, logging, and policy implementation.

Unit 5 Cyber Ethics and Human Factors in Cyber Security 7 (L)hrs

Cyber ethics principles, ethical issues in cyberspace, privacy and data protection concerns, psychological aspects of attackers, user and victim behavior, principles and responsibilities of ethical hacking, social engineering awareness and prevention, organizational security culture and human responsibility.

Unit 6 Case Studies 6 (L)hrs

Real-world cyber security case studies, analysis of cyber-attacks, lessons learned from incidents, preventive strategies, legal and ethical implications of major cyber security incidents.

Laboratory Work

In the laboratory, students will perform hands-on practical exercises using Wireshark, web browser–based tools etc. to analyze cyber security concepts. Laboratory experiments are aligned with theory and designed for real-world applications.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Study of Cybercrime Scenarios and Associated Impacts.
2.	Study a cybercrime case related to the IT Act, 2000 and identify applicable sections.
3.	Analysis of Active Phishing Attacks and Identification of Fake Websites.
4.	Password Strength Evaluation and Analysis Using Browser-Based Tools.
5.	Installation and Configuration of Wireshark on Windows and Linux Platforms
6.	Observation and Analysis of Active Denial-of-Service Attack Traffic Patterns Using Wireshark.
7.	Comparative Analysis of Secure and Insecure Web Communication (HTTP vs HTTPS).
8.	Packet Capture and Network Traffic Analysis Using Wireshark.
9.	Observation and Analysis of SSL/TLS Encrypted Network Traffic.
10.	Protocol Analysis of HTTP, DNS, and TCP Using Wireshark.
11.	Network Log Monitoring and Incident Detection Using Wireshark.
12.	Study of Social Engineering Techniques and Cyber Ethics Awareness.
13.	Case Study Analysis of Real-World Cyber Security Incidents

Text Books

1. Sunit Belapure and Nina Godbole, “Cyber Security: Understanding Cyber Crimes”, Computer Forensics and Legal Perspectives, Wiley India, 1st edition, 2011.
2. Thomas A. Johnson, “Cyber Security”, CRC Press, 1st edition, 2015.
3. Eric Cole, “Cyber Security Essentials”, Wiley, 1st edition, 2010.

Reference Books

1. William Stallings and Lawrie Brown, “Computer Security: Principles and Practice”, Pearson Education, 4th edition, 2018.
2. Charles P. Pfleeger and Shari Lawrence Pfleeger, “Security in Computing', Prentice Hall, 5th edition, 2015.

Online Learning Material

1. https://www.coursera.org/learn/introduction-to-cybersecurity-fundamentals

Accessed on 14-12-2025

2. https://www.nist.gov/cyberframework Accessed on 14-12-2025
3. https://owasp.org/www-project-top-ten/ Accessed on 14-12-2025
4. https://www.cloudflare.com/learning/security/ Accessed on 14-12-2025
5. https://www.cisa.gov/cybersecurity Accessed on 14-12-2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Introduction to Cyber Security	Dr. Jeetendra Pande	Uttarakhand Open University (via SWAYAM)	https://onlinecourses.swayam2.ac.in/nou19_cs08/preview
2	Cyber Security	Dr. G. Padmavathi	Avinashilingam Institute (via SWAYAM)	https://onlinecourses.swayam2.ac.in/cec23_cs16/preview
3	Cyber Security, Tools, Techniques and Counter Measures	Prof. Nilesh K. Modi	Dr. Babasaheb Ambedkar Open University (via SWAYAM)	https://onlinecourses.swayam2.ac.in/nou25_ge62/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS112
Course Title: Software Engineering

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 5%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply various software process models and agile methods to software development problems.
2	Analyze requirements and represent decision logic using techniques like decision trees and tables.
3	Create design diagrams such as DFDs, ER diagrams and structure charts based on software design principles.
4	Evaluate project management tools including metrics, estimation methods, scheduling and risk plans.
5	Apply testing techniques including unit, integration and system testing to various software problems.
6	Analyze version control systems and software maintenance approaches for managing software evolution.

Contents

Part-A

Unit-1 Introduction to Software Engineering **3(L) hrs**

Nature of Software, Software Engineering, Software Process, Software Myths, Software application domains, Software crisis – Problem and causes.

Unit-2 Software Development Lifecycle **10(L) hrs**

Software Process Models - Waterfall model, Prototype model, Spiral model, Incremental model, Evolutionary model, Agile model. Introduction to Software Development Methodologies – Agile development - Scrum, XP. DevOps principles (CI/CD).

Unit-3 Requirement Engineering **9(L) hrs**

Problem Analysis, Requirement Engineering Process, Requirement Elicitation, Requirement Specification and Validation, documenting Software Requirement Specification (SRS), Requirement Analysis Approaches- Structured Analysis vs Object Oriented Analysis, Functional and non-functional requirements, Techniques for representing complex logic – Decision trees, Decision tables.

Part-B

Unit-4 Software Project Management **8(L) hrs**

Project Management Concepts – Managing People, Product, Process and Project. Process and Project Metrics – Size oriented metrics, Function oriented metrics, Empirical estimation, Heuristic estimation and Analytical estimation, Project monitoring and control– Work breakdown structure, Activity chart, Gantt charts, PERT charts, Critical Path Method, Risk management- Identification, Analysis, Planning and Monitoring, RMMM Plan.

Unit-5 Software Design and Coding **7(L) hrs**

Software Design Principles, Software design concepts – Abstraction, Architecture, Modularity, Functional Independence, Refinement, Refactoring, Function oriented design – Flow modeling through Data Flow Diagram and Data Dictionary, Data Modeling through E-R Diagram, Structure chart, Object-Oriented Design Concepts – objects and object classes. Coding standards, code inspection, code walkthrough and code reviews.

Unit-6 Software Testing and Maintenance **8(L) hrs**

Introduction to Software Testing, Types of Software Testing - Unit Testing, Integration Testing, System Testing, Test Planning and Test Cases, Test Automation, Software Verification and Validation, Software maintenance, Software re-engineering, Reverse engineering, Forward engineering, Challenges in Legacy Systems, Software Updates and Version Control.

Laboratory Work

In the laboratory work, students should implement these concepts using any open-source tools. Students are also required to choose a suitable project and perform the following practical's on the same.

Following is only the suggested list of Practical's. Instructor may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Description of the project (Title, Introduction, Purpose etc.)
2.	Conduct feasibility study for the project.
3.	Create a Software Process Lifecycle Diagram using tools like draw.io/Lucid chart etc. for the project chosen and justify why the chosen SDLC model is appropriate.

4.	Preparation of Software Requirement Specification (SRS) document by clearly specifying all functional and non-functional requirements.
5.	Draw Decision Trees and Tables using draw.io or similar software for a sample logic problem of the project.
6.	Study of OpenProj / Project Libre or similar software for Software Project Management.
7.	Implement Activity diagram and Resource usage of the project using OpenProj / Project Libre or similar software.
8.	Implement Gantt Chart of the project using OpenProj / Project Libre or similar software. Also find critical path.
9.	Study and implement any function-oriented design tool(s) for making Data Flow Diagrams (DFD).
10.	Create ER diagram for the project problem using any open-source tool like draw.io/ dbdiagram.io etc.
11.	Study and implement any function-oriented design tool(s) for making Structure Chart.
12.	Compute test cases using equivalence class partitioning and boundary value analysis for the project sub-problems.
13.	Study of testing tools like Selenium/ Appium/ Postman etc. to understand the testing process.
14.	Perform functional testing of any website like IRCTC, bookmyshow etc. to test search/login and other features and document the observations.

Text Books

1. Roger Pressman and Bruce Maxim, "Software Engineering: A Practitioner's Approach", 9th edition by, McGraw-Hill Education, 2023.
2. Rajib Mall, "Fundamentals of Software Engineering", Prentice Hall of India, 5th edition, 2018.
3. Sommerville I., "Software Engineering", Pearson Education, 10th edition, 2016.

Reference Books

1. P. Jalote, "An Integrated Approach to Software Engineering", Springer, 3rd edition, 2005.
2. B. Hughes, M. Cortell, R. Mall, "Software Project Management", Tata McGraw Hill, 5th edition, 2009.

Online Learning Materials

1. https://nptel.ac.in/courses/10610508 Accessed on 20-05-2025
2. https://www.scrum.org/learning-series/what-is-scrum/ Accessed on 20-05-2025
3. https://www.guru99.com/software-testing.html Accessed on 20-05-2025
4. https://www.youtube.com/watch?v=T0TynxN77oY Accessed on 20-05-2025
5. https://www.youtube.com/watch?v=LA_N8cvmLBI Accessed on 20-05-2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Software Engineering	Prof. Rajib Mall	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc20_cs68/preview
2	Software Project Management	Prof. Rajib Mall, Prof. Durga Prasad Mohapatra	IIT Kharagpur, NIT Rourkela	https://onlinecourses.nptel.ac.in/noc22_cs107/preview
3	Software Engineering	Prof. N.L. Sarda, Prof. Umesh Bellur, Prof. R.K. Joshi	IIT Bombay	https://nptel.ac.in/courses/106101061

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS113
Course Title: Design and Analysis of Algorithms

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 90%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): CCS102

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Evaluate the time and space complexity of recursive and non-recursive algorithms.
2	Apply divide-and-conquer strategies to construct efficient algorithmic solutions.
3	Implement Breadth-First Search (BFS) and Depth-First Search (DFS) techniques for solving graph-based problems.
4	Solve optimization problems using appropriate greedy algorithm strategies.
5	Design Dynamic Programming solutions to solve complex optimization problems.
6	Optimize solution strategies for NP-complete problems using reductions and approximation algorithms.

Contents

Part-A

Unit-1 Introduction **4(L) hrs**

Algorithm design and representation, Performance analysis- Space complexity, Time complexity, Asymptotic Notations- Big-Oh (O), Omega (Ω), Theta (Θ), and Little-oh (ο), Little Omega (ω) Notation, Mathematical analysis of non-recursive and recursive algorithms with examples.

Unit-2 Divide and Conquer **5(L) hrs**

General method, solving recurrences using recurrence trees, repeated substitution, statement of Master Theorem, applications – Binary search, Merge sort, Quick sort, Strassen's Matrix Multiplication, Finding the maximum and minimum.

Unit-3 Applications of Graph Traversal Techniques **6(L)hrs**

Representation of graphs, Breadth-First Search method for Single Source Shortest Path on unweighted graphs, Depth First Search using queue, types of connected, biconnected and strongly connected components in directed graphs, topological sorting using Directed Acyclic Graph (DAG).

Unit-4 Greedy Algorithms **7(L)hrs**

Greedy choice, optimal substructure property, minimum spanning trees-Prim's and Kruskal's, shortest path using Dijkstra, Bellman Ford, fractional knapsack, Job Sequencing with Deadline, Activity Selection Problem, Travelling salesperson problem and Huffman coding.

Part-B

Unit-5 Dynamic Programming **7(L)hrs**

Introduction to dynamic programming and application of the algorithm to solve multistage graphs with Forward and backward approach, edit distance, matrix chain multiplication, All pairs shortest path problem using Floyd-Warshall algorithm and 0/1 Knapsack problem.

Unit-6 String Matching **5(L)hrs**

Introduction, Brute Force algorithm, Rabin-Karp algorithm using Hashing Technique, KMP algorithm with π Table, Boyer-Moore algorithm using Bad Match table and Good Suffix heuristic.

Unit-7 Backtracking **5(L)hrs**

General method, N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles**.**

Unit-8NP Completeness **6(L)hrs**

Classes NP, P, NP- Hard, NP-complete and polynomial time reductions, Introduction to approximation algorithms, Absolute approximations, E-approximations, Travelling salesperson problem with NP-complete, SAT (Boolean Satisfiability Problem) to IND-SET **(**Independent Set Problem**)** reduction. IND-SET to Vertex Cover, 2-approximation algorithm for Vertex Cover.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	A college library system stores roll numbers in both sorted and unsorted formats for different departments. Analyze how the choice between Linear Search and Binary Search affects search efficiency in each case.

2	Write a program to sort the class roll numbers of your class using a merge sort algorithm and determine the time required to sort the elements.
3	Write a program to sort the university roll numbers of your class using the Quick sort method and determine the time required to sort the elements.
4	Write a program to use a queue to store the node and mark it as 'visited' until all its neighbours (vertices that are directly connected to it) are marked. Implement by using BFS and DFS algorithms for a graph.
5	Write a program to solve fractional knapsack using the Greedy algorithm.
6	Implement Prim's and Kruskal's algorithms for a common network and compare their effectiveness in finding the minimum spanning tree.
7	Design a program that models a road network and finds the shortest path from home to college using Dijkstra's and Bellman–Ford algorithms.
8	Write a program to find the shortest path of the multistage graph using dynamic programming.
9	Implement the Floyd–Warshall algorithm to compute shortest paths among cities in a state-wide network.
10	Write a program to match the pattern by using Brute Force algorithm and Rabin-Karp algorithm.
11	Write a program to match the String by using KMP algorithm and Boyer-Moore algorithm.
12	Write a program to find the solution to the 8-queens problem using backtracking.
13	Write a program to solve subset sum problems using backtracking.

Text Books

1. Ellis Horowitz, Sartaj Sahni and S. Rajasekharan, "Fundamentals of Computer Algorithms", Universities Press, 2nd Edition, 2007.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms", PHI, 3rd Edition, 2009.
3. P. H. Dave, H. B. Dave, "Design and Analysis of Algorithms", Pearson Education, 2nd Edition, 2007.

Reference Books

1. M. T. Goodrich and R. Tomassia, "Algorithm Design: Foundations, Analysis and Internet examples", John Wiley and sons, 1st Edition, 2001.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford Univ. Press, 2nd Edition, 2014.
3. Aho, Ullman and Hopcroft, "Design and Analysis of algorithms", Pearson Education, 2nd Edition, 2004.
4. R. Neapolitan and K. Naimipour, "Foundations of Algorithms", Jones and Bartlett Student edition, 4th edition, 2010.

Online Learning Materials

1. https://www.coursera.org/learn/cpsc-8400-design-and-analysis-of-algorithms Accessed on Dec 27, 2025
2. https://www.classcentral.com/course/youtube-computer-sc-design-analysis-of-algorithms-47664 Accessed on Dec 27, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Design and analysis of algorithms	Prof. Madhavan Mukund	Chennai Mathematical Institute	https://onlinecourses.nptel.ac.in/noc26_cs67/preview
2	Fundamental Algorithms: Design and Analysis	Prof. Sourav Mukhopadhyay	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc26_cs42/preview
3	Design and Analysis of algorithms	Dr. Faheem Syeed Masoodi	University of Kashmir	https://onlinecourses.swayam2.ac.in/cec20_cs03/preview

Guru Nanak Dev Engineering College, Ludhiana
An Autonomous College under UGC Act 1956
B.Tech. (CSE), Scheme - 2024

Course Code: ESCS102
Course Title: Design Thinking

Programme: B.Tech.	**L:** 0 **T:** 0 **P:** 2	**Credits:** 1
Semester: 5	**Theory/Practical/Integrated:** Practical	**Teaching Hours:** 30(P)=30hrs
Total Max. Marks:50	**Continuous Assessment (CA) Marks:** 50	**End Semester Examination (ESE) Marks:** 0
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 0		
Duration of End Semester Examination (ESE): NIL		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Identify user-centric problems using Design Thinking principles.
2	Apply Design Thinking tools to empathize with users and define problems.
3	Generate and evaluate innovative ideas using structured ideation techniques.
4	Develop and refine prototypes through iterative Design Thinking models.
5	Present ethical and sustainable solutions using effective storytelling and documentation.

Contents

Unit-I Overview of Design Thinking **4(P) hrs**

Introduction to Design Thinking- drivers, attributes, principles of design thinking, 3Ps of design thinking, features and applications of design thinking, design thinking vs traditional thinking. Case Studies- General, Engineering and Service applications.

Activities: Identify an Opportunity and Scope of the Project Explore the possibilities and Prepare design brief

Unit-II Design Thinking tools and techniques **7(P) hrs**

Design Thinking persona, visualisation, stakeholder mapping, journey mapping, mind mapping, Methods and Tools for Empathize and Define phases, star bursting, divergent thinking, convergent thinking, ethnography, brainstorming, storytelling, role playing and user interviews. Six thinking hats.

Activities: Apply Design Thinking Tools and Techniques to Solve a User-Centric Problem.

Unit-III Design Thinking Models **12 (P) hrs**

Double diamond design thinking model- 4D's of double diamond (Discover, Define, Develop and Deliver), feedback mechanism. Standford 5-Phase design thinking model- Empathize, Define, Ideate, Prototype and Test. Feedback and iterations.

Activities:

- Apply the methods of empathize and Define Phases Finalize the problem statement
- Apply the methods of Ideate Phase: Generate lots of Ideas
- Apply the methods of Prototype Phase: Create prototypes for selected ideas
- Collect feedback; iterate and improve the ideas
- Present your solution using Storytelling method

Unit-IV Ethics in Design Thinking and Design Thinking for Sustainability **3 (P)hrs**

Design Thinking Ethics- values and morals, five ways of ethical thinking, ethical issues, universal principles of ethics. Sustainability- three pillars of sustainability, sustainable design, 2030 agenda for sustainable development.

Project Submission: Fine tuning and submission of project report **4 (P)hrs**

Note: Students shall form into groups and identify a problem (preferably societal problem with engineering orientation to solve) suitable for the design thinking and go through the process week-wise. At the end of each phase, brief documentation shall be submitted and a final report covering all phases has to be submitted at the end of the semester.

Text Books:

1. E Balagurusamy, B. Vijaya Kumar, "Design Thinking A Beginners Perspective", McGraw Hill Education India, 1st Edition, 2025.

2. Michael Lewrick, Patrick Link, Larry Leifer, "The Design Thinking Playbook: Mindful Digital Transformation of Teams, Products, Services, Businesses and Ecosystems", Wiley, Edition- May 2018

Reference Books:

1. Michael Lewrick, Patrick Link, Larry Leifer, "The Design Thinking Toolbox: A Guide to Mastering the Most Popular and Valuable Innovation Methods", Wiley, Edition- March 2020

2. Shalini Rahul Tiwari, Rohit Rajendra Swarup, "Design Thinking: A Comprehensive Textbook", Wiley India, 2024.

3. Tim Brown , "Change by Design, Revised and Updated How Design Thinking Transforms Organizations and Inspires Innovation", Harper Collins, 2019

Online Learning Material

1. Design Thinking and Innovation — IIT Bombay (Coursera)
https://www.coursera.org/learn/designthinkingandinnovation Accessed on 09-01-2026

2. Design Thinking for Innovation — University of Virginia (Coursera)
https://www.coursera.org/learn/uva-darden-design-thinking-innovation Accessed on 09-01-2026

3. Design Thinking and Innovation — Harvard Business School Online
https://online.hbs.edu/courses/design-thinking-innovation/ Accessed on 09-01-2026

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Design Thinking and Innovation	Prof. Ravi Poovaiah	IIT Bombay	https://onlinecourses.swayam2.ac.in/aic26_ge05/preview

Semester 6

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS114

Course Title: Cloud Computing

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 10%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): CCS104, CCS105, CCS106

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Discuss cloud computing concepts, service models and cloud deployment models.
2	Apply virtualization technology in cloud implementation.
3	Manage storage and databases on cloud platforms.
4	Implement cloud networking, content delivery network and load balancing in cloud.
5	Analyze security threats and implement security in cloud systems.
6	Compare open source and commercial cloud platforms.

Contents

Part-A

Unit-1 Cloud Computing Fundamentals **7(L) hrs**

Overview of computing paradigms - Distributed computing, Parallel computing, Cluster computing, Grid computing, Utility computing, Cloud computing. Cloud reference model architecture, benefits, challenges and risks in cloud computing. Cloud service models - Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Cloud deployment models - Public, Private, Community, and Hybrid cloud.

Unit-2 Virtualization **8(L) hrs**

Virtualization - Definition, Characteristics and benefits of virtualization, Virtualization and cloud computing, Types of virtualization, Introduction to Amazon Elastic Compute Cloud (EC2) service, Hypervisors, Types of hypervisors. Containerization and Docker, Classic data centre, Virtualized data centre.

Unit -3 Cloud Storage and Databases **8(L) hrs**

Need for cloud storage, Characteristics of cloud storage, Storage as a Service (STaaS), Cloud storage models (Object, Block and File), Introduction to Amazon Web Services (AWS) Simple Storage Service(S3) and Elastic Block Store (EBS), Cloud storage architecture, Cloud databases (Relational and Non-relational), Introduction to AWS Relational Database Service (RDS) and DynamoDB service.

Part-B

Unit-3 Cloud Networking, Content Delivery and Load Balancing **8(L) hrs**

Cloud networking - Networking basics, Virtual Private Cloud (VPC), Introduction to Amazon VPC service. Content delivery- Content Delivery Network (CDN), Introduction to Amazon CloudFront service. Load Balancing: Introduction, Types of load balancing, Introduction to Amazon Elastic Load Balancing (ELB) service.

Unit-5 Cloud Security **7(L) hrs**

Cloud security issues and challenges, Cloud security shared responsibility model, Cyber threats, Data encryption, Identity and access management: Introduction to AWS IAM service, Digital signature, Secure Socket Layer (SSL), Multi Factor Authentication (MFA).

Unit-6 Emerging Technologies and Cloud Computing Platforms **8(L)hrs**

Emerging technologies: Edge computing, Fog computing and Quantum computing. Study and comparison of open source cloud platforms (OpenStack, OpenNebula, CloudStack etc.) and commercial cloud platforms (Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform etc.).

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Install VirtualBox/VMware and create multiple Virtual Machines (VMs) with Linux/Windows.
2	Create, manage, and deploy Docker containers.
3	Create an Amazon Web Services (AWS) free tier account and explore the services on AWS platform.
4	Create, launch, and monitor a virtual machine instance using cloud platform.
5	Create a block storage volume and attach it to the virtual instance. Configure the instance to use the virtual disk.

6	Build a database server and interact with it using a web application.
7	Build a Virtual Private Cloud (VPC) and launch a web server into the VPC.
8	Use load balancing and auto scaling service to load balance and scale the architecture.
9	Create users, add users to groups and test users using identity and access management service.

Text Books

1. Raj Kumar Buyya, James Broberg, Andrezei Goscinski, "Cloud Computing: Principles and Paradigms", John Wiley & Sons Inc, 1st edition, 2011.
2. Barrie Sosinsky, "Cloud Computing Bible",Wiley India Pvt. Ltd, 1st edition, 2011.
3. John Rittinghouse, James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press Taylor and Francis Group, International edition, 2016.
4. Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper," Cloud Computing for Dummies", John Wiley & Sons Inc, International edition, 2020.

Reference Books

1. Ronald L. Krutz and Russel Dean Vines, "Cloud Security: A Comprehensive Guide to Secure Cloud Computing", Wiley-India Pvt. Ltd, 1st edition, 2010.
2. Thomas Erl, Zaigham Mahmood, Ricardo Puttini, "Cloud Computing Concepts, Technology, and Architecture", Prentice Hall, 1st edition, 2013.
3. Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill, 1st edition, 2010.
4. Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi, "Mastering Cloud Computing", Tata McGrawHill, International edition, 2013.

Online Learning Materials

1. https://www.youtube.com/watch?v=mxT233EdY5c Accessed on 23-12-2025
2. https://www.youtube.com/watch?v=_pPlanX5wQY Accessed on 23-12-2025
3. https://www.youtube.com/watch?v=sF014G2BUXM Accessed on 23-12-2025
4. https://www.youtube.com/watch?v=IYI_Co26Sdo Accessed on 23-12-2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Cloud Computing	Prof. Soumya Kanti Ghosh	IIT Kharagpur	https://nptel.ac.in/courses/106105167

2	Cloud Computing and Distributed Systems	Prof. Rajiv Misra	IIT Patna	https://nptel.ac.in/courses/106104182

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS115

Course Title: Compiler Design

Programme: B.Tech.	**L:** 3 **T:** 1 **P:** 2	**Credits:** 5
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L) + 15(T) + 30(P) = 90 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Analyze end-to-end compilation process, including language processing systems, runtime environments, and symbol management mechanisms.
2	Apply formal language concepts and lexical analysis techniques to design and implement efficient tokenizers using programming languages and generator tools.
3	Construct syntax analysis solutions using top-down and bottom-up parsing methods for grammar-driven language translation.
4	Develop syntax-directed translations and intermediate representations to correctly model expressions, control flow, and procedure calls.
5	Apply target-machine-oriented code generation strategies with effective register allocation and local optimizations.
6	Analyze intermediate and target code using machine-independent optimization techniques based on data-flow and control-flow analysis.

Contents

Part-A

Unit-1 Introduction to Compilers **5(L) hrs**

Language Processors, Structure of a Compiler, Phases of Compiler, Grouping of Phases into Passes, Analysis-Synthesis Model of Compilation, Symbol Table Format, Data Structures for Symbol tables, Hash tables, Activation tree, and Activation record, Applications of Compiler Technology.

Unit-2 Lexical Analysis **8(L) hrs**

Role of lexical analyzer, Tokens, Patterns, and Lexemes, Attributes for Tokens, Lexical Errors, Input Buffering, Sentinels, Specification of Tokens, Recognition of Tokens, The Lexical Analyzer Generator - LEX.

Unit-3 Syntax Analysis and Syntax Directed Translations **10(L) hrs**

Introduction, Role of the Parser, Context Free Grammars (CFG), Writing a Grammar, Top Down Parsing, Backtracking, LL(1), Recursive Descent Parsing, Non-Recursive Predictive Parsing, Bottom Up Parsing, Shift Reduce Parsing, LR Parsers, SLR Parser. Canonical LR Parser, LALR Parser, Syntax Directed Translation, Syntax Directed Definitions (SDD), L-Attributed SDD's, Parser Generator – YACC.

Part-B

Unit-4 Intermediate Code Generation **8(L)hrs**

Variants of Syntax Trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Conversion, Type Checking, Control Flow, Backpatching, Switch Statements, Intermediate Code for Procedures**.**

Unit-5 Code Generation **8(L) hrs**

Issues in designing a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks, Flow Graphs, DAG's, A Simple Code Generator, Register Allocation and Assignment.

Unit-6 Machine-Independent Optimizations **6(L) hrs**

The Principal Sources of Optimization (Dead Code Elimination, Common Sub-expression Elimination, Constant propagation, Strength Reduction, Loop Optimization), Peephole Optimization, Loops in Flow Graphs, Data Flow Analysis.

Laboratory Work

Students will perform the laboratory experiments by writing and executing programmes in C++.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	To implement Symbol Table using structures.
2	To identify whether a given line is a comment or not.
3	To recognize strings under ***'a', 'a*b+', 'abb'.***
4	To test whether a given identifier is valid or not.

5	To simulate lexical analyzer for validating operators.
6	To implement Recursive Descent Parser for a given expression.
7	To demonstrate implicit and explicit Type Conversion.
8	To generate intermediate code for if–else Statement.
9	To generate intermediate code for Switch-case.
10	To build a DAG for the given expression.
11	To optimize code by removing Common sub-expressions from a sample code.
12	To implement a peephole optimizer that removes redundant instructions from intermediate code.

Text Books

1. Alfred V. Aho, Monica S. Lam, Ravi Seth, Jeffrey D. Ullman, "Compilers, Principles, Techniques, & Tools", Pearson Education, Inc., 2nd edition, 2007.
2. Keith D. Cooper & Linda Torczon, "Engineering a Compiler", Morgan Kaufmann Publishers, 3rd edition, 2022.

Reference Books

1. Manoj B Chandak and Khushboo P Khurana, "Compiler Design", Orient Blackswan Pvt Ltd Ist edition, 2018.
2. S. Chattopadhyay, "Compiler Design", PHI Learning, 2nd edition, 2022.
3. C. Holub., "Compiler Design in C", Prentice-Hall Inc., 2nd edition, 1993.
4. Andrew W Appel., "Modern Compiler Implementation in C++", Cambridge University Press Revised edition, 2004.

Online Learning Materials

1. https://www.geeksforgeeks.org/compiler-design/compiler-design-tutorials/ Accessed on Dec. 17, 2025.
2. https://www.youtube.com/watch?v=5ZmFlxrNaN8 Accessed on Dec. 17, 2025
3. https://www.classcentral.com/course/compilers-328 Accessed on Dec. 17, 2025
4. https://www.edx.org/learn/compilers Accessed on Dec. 17, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Compiler Design	Prof. Santanu Chattopadhyay	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc26_cs56/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: CCS116
Course Title: Machine Learning

Programme: B.Tech.	**L:** 3 **T**: 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): BSCS101, CCS108

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Analyze machine learning problems, machine learning process and evaluation techniques.
2	Implement supervised learning algorithms for regression and classification tasks.
3	Apply unsupervised learning methods for clustering and dimensionality reduction.
4	Develop reinforcement learning solutions for sequential decision-making problems.
5	Design ensemble and tree-based learning models for predictive analytics.
6	Construct neural network and Bayesian models for complex learning problems.

Contents

Part-A

Unit-1 Introduction **6(L) hrs**

Defining Machine Learning, Types of Machine Learning, Well-posed Learning Problem, Basic Terminology: Dataset, features, labels, instances, model, training, validation, testing, prediction, evaluation metrics. Machine Learning Process. Perspectives and Issues in Machine Learning.

Unit-2 Supervised Learning **7(L) hrs**

Regression problem, Linear regression model: Describe relationships, predicting future outputs, Classification problem, Logistic Regression: Learning the logistic regression from training data, Decision boundaries for logistic regression, k-Nearest Neighbor, Decision Trees, Naive Bayes, Support Vector Machines, Nonlinearity and kernel Methods.

Unit-3 Unsupervised Learning **7(L) hrs**

Clustering, k-means clustering, kernel methods, Dimensionality Reduction: Principal Component Analysis (PCA).

Unit-4 Reinforcement Learning **6(L) hrs**

Introduction, The Learning task, Q Learning, Nondeterministic rewards and actions, Temporal difference learning, Generalizing from examples.

Part-B

Unit-5 Decision Tree Learning **6(L) hrs**

Introduction, decision tree representation, appropriate problems for decision tree learning, the basic decision tree learning algorithm, hypothesis space search in decision tree learning, Ensemble methods-Bagging, Gradient Boosting, Random Forest.

Unit-6 Artificial Neural Networks **7(L) hrs**

Introduction, Neural network representation, appropriate problems for neural network learning, perceptron, gradient descent and the delta rule, Multilayer networks, Derivation of Back propagation rule, back propagation algorithm.

.

Unit-7 Bayesian Learning **6(L) hrs**

Bayes theorem and concept learning, Brute- Force Bayes Concept Learning, Maximum likelihood and least squared error hypothesis for predicting probabilities, Bayes optimal classifier, Naive Bayes classifier, Bayesian belief networks.

Laboratory Work

Students will implement the laboratory experiments using open-source programming tools on local or cloud-based environments.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	To set up the Python environment and perform basic data handling and data preprocessing on a given dataset.
2	To implement and analyze a linear regression model for predicting continuous target values and evaluate it.
3	To build a logistic regression classifier for and study decision boundaries and classification performance using suitable metrics.

4	To implement and evaluate the k-Nearest Neighbour classifier and analyze the effect of different values of k on classification accuracy.
5	To apply k-means clustering on unlabeled data and determine the optimal number of clusters.
6	To apply Principal Component Analysis (PCA) for dimensionality reduction and visualize high-dimensional data in lower dimensions.
7	To implement Q-learning for a simple reinforcement learning environment and study agent learning behavior through reward convergence plots.
8	To construct and visualize a decision tree classifier and analyze its performance on a labeled dataset.
9	To implement a single-layer perceptron for binary classification.
10	To design and train a multilayer feedforward neural network using backpropagation and study the effect of hidden layers, learning rate, and epochs on model performance.
11	To implement the Naive Bayes classifier for probabilistic classification and evaluate its performance on numerical or text dataset using standard evaluation metrics.
12	To construct a Bayesian belief network, perform probabilistic inference, and analyze conditional dependencies among variables in a real-world problem domain.

Text Books

1. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Education, 1st Edition, 1997.
2. Wolf, Andrew, “Machine Learning Simplified: A Gentle Introduction to Supervised Learning”, 1st Edition, 2022.
3. Harrington, Peter, “Machine Learning in Action”, 1st Edition, Manning Publications, Manning Publications, 1st Edition, 2012.
4. Sutton, Richard S., and Barto, Andrew G., “Reinforcement Learning: An Introduction”, MIT Press, Cambridge, MA, 2nd Edition, 2018.

Reference Books

1. Bishop, Christopher M., “Pattern Recognition and Machine Learning”, Springer, 1st Edition, 2006.
2. Müller, Andreas C. & Guido, Sarah, “Introduction to Machine Learning with Python: A Guide for Data Scientists”, Shroff/O’Reilly, 1st Edition, 2016.
3. Shalev-Shwartz, Shai & Ben-David, Shai, “Understanding Machine Learning: From Theory to Algorithms”, Cambridge University Press, 1st Edition, 2014.

4. Alpaydin, Ethem, "Introduction to Machine Learning", MIT Press, 4th Edition, 2020.

Online Learning Materials

1. Introduction to Machine Learning. nptel.ac.in/courses/106106139 Accessed on 10 January 2026.
2. Introduction to Machine Learning. nptel.ac.in/courses/106105152 Accessed on 10 January 2026.
3. Machine Learning for Engineering and science applications. nptel.ac.in/courses/106106198 Accessed on 10 January 2026.

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Introduction to Machine Learning	Prof. Balaraman Ravindran	IIT, Madras	nptel.ac.in/courses/106106139
2	Introduction to Machine Learning	Prof. Sudeshna Sarkar	IIT, Kharagpur	nptel.ac.in/courses/106105152
3	Machine Learning for Engineering and science applications	Prof. Balaji Srinivasan Prof. Ganapathy Krishnamurthi	IIT, Madras	nptel.ac.in/courses/106106198

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: FSCS101

Course Title: Corporate Readiness and Employability Enhancement Program

Programme: B.Tech.	**L:** 2 **T:** 0 **P:** 0	**Credits:** 2
Semester: 6	**Theory/Practical:** Theory	**Teaching Hours:** 30(L) = 30 hrs
Total Max. Marks: 100	**Continuous Assessment (CA) Marks:** 40	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Core Course		

Prerequisites (if any): NIL

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Develop a professional digital identity by creating resumes, portfolios, and professional profiles on networking platforms
2	Apply AI-powered tools and digital productivity platforms to support learning, problem solving, and professional tasks.
3	Analyze real-world problems using computational and analytical thinking approaches to identify logical and structured solutions
4	Interpret and analyze data using basic quantitative and logical reasoning techniques for informed decision making.
5	Construct clear and professional written communication**,** including emails and structured content, using appropriate grammar and comprehension techniques
6	Execute effective communication and collaboration strategies during group discussions, presentations, and interview situations

Contents

Part-A

Unit-1 Digital Professional Identity and Personal Branding 4(L) hrs

Introduction to Professional Identity in the Digital Era, Personal Branding and Career Positioning, Resume Writing and ATS Awareness (overview), Professional Networking using LinkedIn and other platforms, Building Digital Portfolios and GitHub Profiles, Managing Digital Footprint and Online Reputation, Professional Email Communication and Etiquette

Unit-2 AI Tools and Digital Productivity **8(L)hrs**

Introduction to Artificial Intelligence in Professional Work, Overview of Generative AI Tools for Learning and Productivity, Prompt Engineering Basics for Effective AI Interaction, AI Tools for Writing, Presentations, and Research, AI-assisted Coding Tools and Developer Productivity, Ethical Use of AI and Academic Integrity, Digital Productivity Platforms (Notion, Trello, Google Workspace), Time Management and Smart Work Techniques using Digital Tools

Part-B

Unit-3 Analytical Thinking and Problem Solving **6(L) hrs**

Introduction to Computational Thinking, Structured Problem Solving Approaches, Analytical Reasoning and Logical Thinking, Data Interpretation using Graphs and Charts, Basic Quantitative Reasoning for Decision Making, Real-life Case Studies on Problem Solving in Industry

Unit-4 Workplace Communication and Industry Readiness **6(L)hrs**

Professional Communication Skills, Verbal and Non-verbal Communication, Presentation Skills and Public Speaking, Teamwork, Collaboration, and Leadership, Workplace Ethics and Professional Behavior, Group Discussions and Structured Thinking, Interview Preparation Strategies and Behavioral Interviewing

Text Books

1. Stephen R. Covey, “The 7 Habits of Highly Effective People”, 30th Anniversary Edition, Simon & Schuster, 2020.
2. Andrew Ng, “AI for Everyone”, DeepLearning.AI, Coursera Course Material Reference (2019).
3. Cal Newport, “Deep Work: Rules for Focused Success in a Distracted World”, Grand Central Publishing, 2016.
4. RS Aggarwal, ‘’A Modern Approach to Verbal and Non - Verbal Reasoning’’ Second Edition, S.Chand, 2018.

Reference Books

1. Melonie Dodaro, “LinkedIn Unlocked: Unlock the Mystery of LinkedIn to Drive More Sales Through Social Selling”, Second Edition, CreateSpace Independent Publishing (2023).
2. Ethan Mollick, “Co-Intelligence: Living and Working with AI”, Portfolio / Penguin Random House (2024).
3. Chip Heath and Dan Heath, “Made to Stick: Why Some Ideas Survive and Others Die”, Random House (2007).
4. Daniel Goleman, “Emotional Intelligence: Why It Can Matter More Than IQ”, Bantam Books (2005).

Online Learning Material

1. https://www.geeksforgeeks.org/questions/aptitude/
2. https://www.geeksforgeeks.org/dsa/fundamentals-of-algorithms/
3. https://bookboon.com/
4. https://ocw.mit.edu/
5. https://www.udemy.com/
6. Enhancing Soft Skills and Personality, NPTEL, IIT Kanpur, https://onlinecourses.nptel.ac.in
7. Developing Soft Skills and Personality, NPTEL, IIT Kanpur, https://onlinecourses.nptel.ac.in
8. Soft Skills Development, NPTEL, IIT Kharagpur, https://nptel.ac.in
9. AI for Everyone, Coursera (DeepLearning.AI), https://www.coursera.org/learn/ai-for-everyone
10. Career Success Specialization, Coursera, https://www.coursera.org/specializations/career-success
11. Digital Skills: Digital Skills for Work and Life, Future Learn, https://www.futurelearn.com
12. Communication Skills for Engineers, NPTEL, https://onlinecourses.nptel.ac.in
13. Introduction to Artificial Intelligence, Microsoft Learn, https://learn.microsoft.com
14. Google Digital Garage – Fundamentals of Digital Marketing and Productivity, https://learndigital.withgoogle.com
15. LinkedIn Learning – Career Development and Professional Skills Courses, https://www.linkedin.com/learning

Supplementary SWAYAM Course

Sr. No	Course Name	Instructor	Host Institute	URL
1	Soft Skill Development	By Prof. Priyadarshi Patnaik, Prof. V.N. Giri, Prof. D. Suar	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc23_hs10/preview
2	Business Communication	By Prof. K Preetha,	JAIN(Deemed-to-be University), Bengaluru	https://onlinecourses.swayam2.ac.in/imb24_mg54/preview
3	Campus Recruitment Training	By Mr. K Preetham	JAIN (Deemed-to-be University), Bangalore	https://onlinecourses.swayam2.ac.in/imb25_mg210/preview
4	Logic And Sets	By Mr. Mohamed Nishad	Farook College, Kozhikode	https://onlinecourses.swayam2.ac.in/cec23_ma22/preview

		Maniparambath		
5	Data Structures and Algorithms Design	By Prof. Nitin Saxena	IIT Kanpur	https://onlinecourses.swayam2.ac.in/cec23_ma22/preview
6	Computer Programming	By Dr. T. Sugirtha	IIIT Tiruchirappalli	https://onlinecourses.nptel.ac.in/noc22_cs91/preview
7	Developing Soft Skills and Personality	By Prof. T. Ravichandran	IIT Kanpur	https://onlinecourses.nptel.ac.in/noc22_hs77/preview

Professional Elective Courses

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS101
Course Title: Generative AI

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 60%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – I		

Prerequisites (if any): CCS108

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Explain the principles of Generative AI and differentiate generative and discriminative models with suitable examples.
2	Apply probability distributions, sampling methods, and basic language models to represent and generate data.
3	Design and analyze generative models such as Autoencoders, Variational Autoencoders, and Restricted Boltzmann Machines for synthetic data generation.
4	Analyze the architecture, training process, and applications of generative adversarial networks.
5	Analyze and evaluate transformer-based generative models and large language models for text and multimodal generation tasks.
6	Evaluate generative model outputs using standard metrics and assess ethical and responsible AI considerations.

Contents

Part-A

Unit-1 Introduction to Generative AI **7 (L) hrs**

Overview of Artificial Intelligence - comparison of symbolic AI and data-driven AI approaches, Machine Leaning fundamentals - supervised, unsupervised, and semi-supervised learning paradigms with examples, Evolution of Generative AI, Difference between Discriminative and Generative Models, Mathematical intuition and examples, Applications of Generative AI in industry.

Unit-2 Generative Models **8 (L) hrs**

Basics of Probability Distributions - Gaussian, Bernoulli, Multinomial, and their relevance, Sampling Methods - random sampling, Monte Carlo sampling, and importance sampling, Language Models - sequence modeling, probability of word sequences, and generation process,

Variational Autoencoders; Encoder–decoder architecture, Latent variable modeling, Restricted Boltzmann Machines - energy-based models, visible and hidden units, and learning process with case studies.

Unit-3 Generative Adversarial Networks **8 (L) hrs**

Architecture of GANs; Training GANs - loss functions, convergence behavior, and training instability issues, Challenges in GAN training - mode collapse, vanishing gradients, and imbalance between networks. Types of GANs - Deep Convolutional GANs, CycleGAN, StyleGAN; Applications of GANs in image and video generation, deepfakes, media synthesis, and industrial case studies.

Part-B

Unit -4 Transformer Architecture and Attention Mechanisms **7 (L) hrs**

Limitations of RNNs and CNNs for sequence modeling, Introduction to attention mechanisms and self-attention, scaled dot-product attention and intuition, multi-head attention and contextual representation learning, Positional encoding and sequence order representation, Transformer encoder–decoder architecture, Autoregressive generation using transformer models.

Unit -5 Large Language Models and Multimodal Generative Systems **8 (L) hrs**

Transformer-based language modeling, GPT architecture: training objective, text generation process, and applications, BERT architecture - masked language modeling and bidirectional representations, Comparison of BERT and GPT, Large Language Models (LLMs) - scaling laws, pretraining, fine-tuning, and instruction tuning. Multimodal generative models - text-to-image generation, Overview of DALL-E and Stable Diffusion architectures.

Unit -6 Applications, Challenges and Ethics **7(L) hrs**

Applications in text, audio, video, and design; Evaluation metrics - FID, BLEU, ROUGE; Ethical implications, bias, misinformation, copyright issues, Responsible AI practices.

Laboratory Work

In the laboratory work, students will implement programs related to the above topics using open-source programming languages, frameworks, and libraries such as Python, NumPy, and related tools. Cloud-based environments like Google Colab may be used for execution.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Study real-world AI applications and classify them as generative or discriminative models
2	Analyze the evaluation of modern LLMs for different tasks using standard benchmarks

	and metrics
3	Analyze probability distributions (Gaussian, Bernoulli, Multinomial) and demonstrate their role in generative data modeling
4	Implement and compare different sampling techniques (random, Monte Carlo, importance sampling) to study diversity and quality of generated data
5	Design and implement an Autoencoder to learn latent representations and analyze reconstruction performance
6	Implement a Variational Autoencoder and generate new data samples by sampling from the learned latent space
7	Implement a Generative Adversarial Network for image generation and analyze training stability and output quality
8	Study transformer architecture and attention mechanisms by implementing a basic transformer-based text generation model
9	Use an open-source generative model (LLMs or image generation models such as LLaMA, Qwen, DeepSeek, Stable Diffusion) for a chosen application.
10	Evaluate generative AI model outputs using metrics (BLEU, ROUGE, FID) and analyze ethical issues, bias, and responsible AI practices.

Text Books

1. David Foster, "Generative Deep Learning: Teaching Machines to Paint, Write, Compose and Play", O'Reilly Media, 1st edition, 2019.
2. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning", MIT Press, 5th Edition ,2016.

Reference Books

1. Jakub Langr, Vladimir Bok, "GANs in Action: Deep Learning with Generative Adversarial Networks", Manning Publications, 1st Edition, 2019.
2. Papoulis and S. U. Pillai, "Probability – Random Variables, and Stochastic Processes", 4th edition, McGraw-Hill, 2017.

Online Learning Materials

1. CS236 – Deep Generative Models, Stanford University (Lecture Notes & Videos) https://deepgenerativemodels.github.io Accessed on 09-01-2026
2. CS231n – Convolutional Neural Networks for Visual Recognition, Stanford (GAN & Autoencoder sections) https://cs231n.stanford.edu Accessed on 09-01-2026
3. DeepLearning.AI – Generative AI Specialization, Andrew Ng https://www.deeplearning.ai

Accessed on 09-01-2026

4. Hugging Face Documentation (Transformers, LLMs, Diffusion Models)
   https://huggingface.co/docs Accessed on 09-01-2026

Supplementary SWAYAM Courses

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Generative AI and Large Language Models	Naveen Kumar Bhansali	IIT Madras	https://onlinecourses.swayam2.ac.in/imb25_mg135/preview
2	Python for Data Analytics, Machine Learning and Generative AI: A Comprehensive Guide	By Dr. S AJITHA	M S Ramaiah University of Applied Sciences	https://onlinecourses.swayam2.ac.in/imb26_mg130/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS102
Course Title: Data Warehousing and Business Intelligence

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – I		

Prerequisites (if any): CCS104

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Analyze data warehousing concepts, components, architectures, and enterprise integrations.
2	Design ETL processes and multidimensional data warehouse schemas for OLAP systems.
3	Apply association rule mining techniques and the Apriori algorithm for knowledge discovery.
4	Examine Business Intelligence architectures, governance, and decision-support roles.
5	Evaluate Business Intelligence solutions using KPIs, BI tools, security measures and cost-benefit analysis.
6	Implement Business Intelligence models for real-world business applications.

Contents

Part-A

Unit-1 Introduction to Data Warehousing **8(L) hrs**

Defining Features - Subject-Oriented Data, Integrated Data, Time-Variant Data, Nonvolatile Data, Data Granularity, Data Warehouses and Data Marts, Top-Down Versus Bottom-Up Approach. Components of Data Warehouse - Source Data Component, Data Staging Component, Data Storage Component, Information Delivery Component, Metadata Component, Management and Control Component. Metadata in the Data Warehouse, Trends in Data Warehousing, Data Warehousing and ERP, Data Warehousing and KM, Data Warehousing and CRM.

Unit-2 Architecture and Infrastructure of Data Warehouse **8(L) hrs**

The Data Warehouse Project, Understanding Data Warehouse Architecture, Architectural Framework, Technical Architecture, Infrastructure Supporting Architecture, Hardware and Operating Systems,

Data Modeling, Data Extraction, Data Transformation, Data Loading, Data Quality, Online Analytical Processing (OLAP). Data Warehouse Schema, Introduction to Data Warehouse Schema - Dimension, Measure, Fact Table, Multi-dimensional view of data, Star Schema, Snowflake Schema, Fact Constellation Schema (Galaxy Schema), Comparison among Star, Snowflake and Fact Constellation Schema.

Unit-3 Introduction to Data Mining **7(L) hrs**

Introduction to Data Mining, Data Mining operations and techniques, The Data Mining Process, The Knowledge Discovery Process, Data Warehousing v/s Data Mining, Defining Association Rule Mining, Representations of Items for Association Mining, The Metrics to Evaluate the Strength of Association Rules, The Apriori Algorithm.

Part-B

Unit-4 Introduction to Business Intelligence **6(L)hrs**

Definitions of BI, A brief history of BI, The Architecture of BI, Styles of BI, The Benefits of BI, Intelligence creation and Use, BI governance.

Unit-5 BI Implementation and Advanced BI **8(L)hrs**

BI user Community, Key performance indicators and performance metrics, KPI identification from business objectives, KPI hierarchy- Strategic vs Tactical vs Operational, Lagging vs Leading KPIs, KPI interpretation (trends, thresholds, exceptions), Common KPI pitfalls (vanity metrics, misalignment), Developing or acquiring BI systems, Justification and Cost-Benefit Analysis, Security and protection of privacy, Major tools and techniques of BI, Connecting BI systems to databases and other enterprise systems, Limitations of traditional BI, Future of BI, Online social networking: Basics and examples, Collaborative Decision Making.

Unit-6 Business Intelligence Applications **8(L) hrs**

Marketing models including Relational Marketing, Sales force Management, Logistic and production models including Supply chain optimization, optimization models for logistics planning, Business case studies such as Retention in telecommunications, Social Media & Brand Sentiment Analysis, E-commerce Funnel & Website Analytics, Customer Segmentation for Business Strategy, Cross-selling in the retail industry, Logistics planning in food industry.

Laboratory Work

Students will implement laboratory experiments using open-source database, ETL, data mining, and BI tools on local or cloud-based environments.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Design and implement a subject-oriented, time-variant, non-volatile data warehouse

	and a data mart**.**
2	Implement Extract, Transform, Load (ETL) operations using SQL.
3	Design and implement Star Schema and Snowflake Schema.
4	Implement a Fact Constellation (Galaxy) Schema and perform OLAP operations.
5	Implement Association Rule Mining concepts using SQL queries.
6	Implement the Apriori Algorithm on a given dataset to discover frequent item sets and association rules using a data mining tool.
7	Understand Business Intelligence architecture and implement data integration from multiple heterogeneous data sources into a unified BI layer.
8	Define, calculate, and visualize Key Performance Indicators (KPIs) using SQL queries and a BI visualization tool.
9	Implement a BI system using cost–benefit analysis, ROI, and payback period**.**
10	Study and implement Business Intelligence techniques for any one of following: i. Relational Marketing ii. Salesforce Management iii. Customer Retention iv. Cross-Selling v. Supply Chain and Logistics Optimization vi. E-commerce Funnel & Website Analytics. vii. Customer Segmentation for Business Strategy

Text Books

1. Efraim Turban, "Business Intelligence: A Managerial Approach", Pearson Publication, 2nd Edition, 2010.
2. Amit Johri, Dr. Akshai Aggarwal, "Business Intelligence", Morgan Kaufmann, 4th Edition, 2017.
3. Paulraj Ponniah, "Data Warehousing Fundamentals for IT Professionals", Wiley, 2nd Edition, 2008.
4. Jiawei Han and Micheline Kamber, "Data Mining Concepts & Techniques", Elsevier Pub, 3rd Edition, 2012.
5. Parteek Bhatia, "Data Mining and Data Warehousing: Principles and Practical Techniques", Cambridge University Press, 1st edition 2019.
6. Reema Thareja, "Data Warehousing", Oxford University Press, 4th edition, 2009.
7. Paulraj Ponniah, "Data Warehousing Fundamentals", John Wiley & Sons, Inc, 2nd Edition,2001.

Reference Books

1. Ralph Kimball, Margy Ross, "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling", Wiley, 3rd edition, 2013.
2. Vikram Pudi, P. Radha Krishana, "Data Mining", Oxford University press, 1st edition, 2009.
3. W. H. Inmon, "Building the Data Warehouse", Wiley, 4th edition, 2009.
4. Margret H. Dunham, "Data Mining: Introductory and Advanced topics", Pearson Education, 4th Edition, 2006.
5. Vikram Pudi, P. Radha Krishana, "Data Mining", Oxford University press, 1st edition, 2009.

Online Learning Materials

1. https://www.coursera.org/courseraplus?query=data+warehousing Accessed on Dec 20, 2025
2. https://www.coursera.org/courses?query=business+intelligence Accessed on Dec 20, 2025
3. https://www.simplilearn.com/free-business-intelligence-course-online-skillup Accessed on Dec 20, 2025
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/business-intelligence-for-beginners Accessed on Dec 27, 2025
5. https://ebooks.lpude.in/computer_application/mca/term_6/DCAP606_BUSINESS_INTELLIGENCE.pdf Accessed on Dec 27, 2025
6. https://anuradhasrinivas.wordpress.com/wp-content/uploads/2013/03/data-warehousing-fundamentals-by-paulraj-ponniah.pdf Accessed on Dec 27, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Introduction to Data Warehousing and Data Mining	Dr. Vaibhav Panwar	Manipal University, Jaipur	https://onlinecourses.swayam2.ac.in/imb25_mg200/preview
2	Data Mining	Prof. Pabitra Mitra	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc21_cs06/preview
3	Business Intelligence & Analytics	By Prof. Saji K Mathew	IIT Madras	https://onlinecourses.nptel.ac.in/noc26_cs64/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS103
Course Title: Cryptography and Network Security

Programme: B.Tech.	**L:** 2 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective-I		

Prerequisites (if any): CCS106

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply appropriate security mechanisms to mitigate common network and wireless security threats, including port-based authentication frameworks.
2	Implement symmetric and asymmetric cryptographic algorithms and analyze their effectiveness for secure communication.
3	Analyze data integrity and authentication mechanisms using cryptographic hash functions, message authentication codes, and digital signatures.
4	Evaluate public key infrastructure, IEEE 802.1X authentication framework, and secure communication protocols for networked and cloud-based systems.
5	Apply cryptographic techniques and security protocols to address challenges in modern network, wireless, and cloud environments.

Contents

Part-A

Unit-1 Introduction **8(L) hrs**

Computer security concepts, security goals, threats and attacks, Security services and mechanisms, OSI security architecture, Fundamental security design principles, Attack surfaces and attack trees, Model for network security and standards, Number theory: divisibility, modular arithmetic, prime numbers, Fermat's theorems, discrete logarithm problem.

Unit-2 Symmetric Key Cryptography **8(L) hrs**

Symmetric cipher model. Classical encryption techniques - substitution and transposition ciphers (Caesar, Playfair, Hill), Block cipher principles, DES: structure, operation, strength and limitations. AES - structure, round transformations, key expansion overview.

Unit-3 Block Cipher Modes and Stream Ciphers **6(L) hrs**

Multiple encryption and Triple DES. Block cipher modes of operation - Electronic Codebook mode, Cipher Block Chaining mode, Cipher Feedback mode, Output Feedback mode, Counter mode, Pseudorandom number generation, Stream ciphers and RC4 with weaknesses.

Part-B

Unit -4 Public Key Cryptography and Data Integrity **8(L) hrs**

Principles of public key cryptography, Symmetric vs asymmetric cryptosystems, RSA algorithm, Diffie–Hellman key exchange. Hash functions - SHA-2 and SHA-3. Message Authentication Codes - HMAC, authenticated encryption (GCM overview). Digital signatures - RSA.

Unit-5 Key Management and Authentication Mechanisms **8(L) hrs**

Key management and distribution, Public Key Infrastructure and X.509 certificates, User authentication mechanisms, IEEE 802.1X authentication framework (supplicant, authenticator, authentication server), Kerberos overview.

Unit-6 Key Network Security Protocols and Cloud Security **7(L) hrs**

Transport Layer Security (TLS), HTTPS and SSH. Wireless security - IEEE 802.11i. Email security - PGP and S/MIME (overview), IP Security (IPsec architecture). Overview of Cloud Security - cloud service models, security challenges. Quantum threats to classical cryptography, Overview of Lattice-based Cryptography.

Laboratory Work

In the laboratory work, students will implement programs related to above topics using any open-source tools, such as Sage Math, OpenSSL, vim, Open LDAP, Python, C, C++, PyCryptodome, ldapadd, ldapsearch, etc.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Apply number theory concepts to cryptographic problems. • Implement modular arithmetic operations • Compute GCD using Euclidean and Extended Euclidean algorithms • Perform modular inverse and Chinese Remainder Theorem calculations **Tools:** SageMath / Python
2.	Apply classical encryption algorithms and analyze their security. • Implement Caesar, Monoalphabetic, Vigenère, and Transposition ciphers • Perform encryption and decryption • Analyze weaknesses using simple cryptanalysis

	Tools: Python / C
3.	Implement and evaluate modern symmetric encryption algorithms. • Implement AES for file/text encryption • Demonstrate different modes of operation (ECB, CBC) • Implement RC4 stream cipher and analyze security issues **Tools:** OpenSSL / Python (PyCryptodome)
4.	Apply asymmetric cryptography for secure communication. • Implement RSA key generation, encryption, and decryption • Demonstrate Diffie–Hellman key exchange • Analyze key sizes and security strength **Tools:** OpenSSL / Python
5.	Analyze integrity and authentication mechanisms. • Generate message digests using SHA • Implement HMAC for message authentication • Create and verify digital signatures using RSA **Tools:** OpenSSL / Python
6.	Apply centralized authentication and access control mechanisms. • Install and configure OpenLDAP server • Create directory entries for users and groups • Integrate LDAP with authentication services (conceptual / demo) • Analyze role of TLS, certificates, and secure authentication **Tools:** OpenLDAP LDAP utilities (ldapadd, ldapsearch)

Text Books

1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson Education, 8th Edition, 2020.
2. Behrouz A. Forouzan and Debdeep Mukhopadhyay, Cryptography and Network Security, McGraw-Hill Education, 2nd Edition, 2012.

Reference Books

1. Atul Kahate, Cryptography and Network Security, McGraw-Hill Education, 3rd Edition, 2017.
2. Eric Rescorla, SSL and TLS: Designing and Building Secure Systems, Addison-Wesley, 1st Edition, 2000.
3. Olaf Kirch, LDAP System Administration, O'Reilly Media, 2nd Edition, 2000.

Online Learning Materials

1. https://www.youtube.com/watch?v=yIm0Ol9Dg4Y&list=PLBlnK6fEyqRgJU3EsOYDTW7m6SUmW6kII&index=4 Accessed on Dec 27, 2025

2. https://www.youtube.com/watch?v=xG3-YcYy0GU&list=PL9FuOtXibFjV77w2eyil4Xzp8eooqsPp8 Accessed on Dec 27, 2025
3. https://www.youtube.com/watch?v=zLNug0LrFiU Accessed on Dec 27, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Cryptography and Network Security	Prof. Sourav Mukhopadhyay	IIT Kharagpur	https://nptel.ac.in/courses/106105162
2	Foundations of Cryptography	Prof. Ashish Choudhury	IIIT Bangalore, IIIT Bangalore	https://nptel.ac.in/courses/106106221
3	Cryptography and Network Security	Dr. Debdeep Mukhopadhyay	IIT Kharagpur	https://nptel.ac.in/courses/106105031

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS104
Course Title: Advanced Computer Networks

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – I		

Prerequisites (if any): CCS106

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Analyze network architectures to ensure the optimal network performance.
2	Implement IOS configurations and TCP/IP protocol concepts to configure and troubleshoot computer networks.
3	Configure various switching mechanisms, STP, and VLANs to build efficient and loop-free switched network topologies.
4	Design optimized internetworking solutions using routing algorithms and protocols.
5	Configure Wireless LANs, IEEE 802.11 operations, and Mobile Ad hoc Networks using network simulation tools to understand wireless communication behaviors.
6	Apply IoT, SDN, and blockchain concepts to utilize modern network architectures effectively.

Contents

Part-A

Unit-1 Internetworking **8 (L) hrs**

Basics, History of Internet, Requirements- perspectives, scalable connectivity, cost effective resource sharing. Network Architecture E-layering and protocols, Internet architecture. Network performance- bandwidth, throughput, latency, (delay). Ethernet Networking - Half and full duplex ethernet, Ethernet at the data link layer, Ethernet at the physical layer, Ethernet cabling - Straight-through, Crossover and rolled cable, Data encapsulation.

Unit-2 Internetworking Operating System (IOS) **4 (L) hrs**

Three-layer hierarchical network model, IOS User Interface- Basics, Router and switch IOS. Command-Line Interface (CLI)- entering, bringing up switch and router, Router and switch administrative configurations- Hostnames, Banners, setting passwords, encrypting passwords.

Unit-3 TCP/IP protocols **8 (L) hrs**

Network Layer Protocols - Internet Protocol (IP)- service model, addressing- classful and classless addressing, subnetting, datagram format, fragmentation, options. Internet Control Message Protocol (ICMP) - messages, debugging tools, ICMP checksum. Dynamic Host Control Protocol (DHCP)- Concept, components, working process and advantages, Network Address Resolution (NAT). Transport Layer Protocols- Transport Layer services, TCP services, features, Segment, TCP connection-connection establishment, data transfer and connection termination. User datagram Protocol (UDP) - services, and application.

Part-B

Unit-4 Switching Architecture and VLANs **8 (L) hrs**

Switching- Overview of switch, Types - Unmanaged and managed switches, functions of switches, switching vs Bridging, limitations of switches, spanning tree protocol (STP), switch administrative configurations- viewing, saving and erasing configurations. Virtual Local Area Network (VLAN) basics - Static VLAN, Dynamic VLAN, Frame tagging, Routing between VLANs, Configuring VLANs.

Unit-5 Routing and Internetworking **8 (L) hrs**

Overview of router, routing basics, Static and dynamic routing, configuring IP routing in network. Inter-domain Routing - Distance vector routing Protocols, Router administrative configurations, Router interfaces, viewing, saving and erasing configurations. Routing Information Protocol (RIP)- RIP implementation, messages, algorithm, timers in RIP and RIP performance. OSPF (Open Shortest Path First) - features, link state advertisement, implementation, messages, algorithm and performance. Inter-domain Routing Protocols - Border Gateway Protocol. (BGP)- features, operation, messages and performance.

Unit-6 Wireless LANs **3 (L) hrs**

Wireless LANs - Introduction, architecture comparison, characteristics and access control. IEEE 802.11 - architecture, MAC Sublayer, Physical layer. Bluetooth- architecture and its layers. Wireless Ad hoc Networks - Mobile Ad hoc Networks (MANETs) - features, advantages, routing in MANETs, applications of MANETs.

Unit-7 Recent Trends in Networking **6 (L) hrs**

Recent trends in networking- green networking, social networks, Internet of Things (IoT) - basics, characteristics and components, IoT vs M2M, IoT Devices and Sensors, Sensor vs Actuator, Types of sensors - temperature, pressure, motion, light, gas. Software Defined Networking- Architecture- Control plane, Data Plane and Application Plane, Interfaces - northbound and southbound. Open Flow Protocol, Flow Tables, Software-Defined Networking (SDN) Operation Workflow, Advantages of SDN. Blockchain - Introduction to Blockchain, Evolution of digital currency. Blockchain technology - Definition and features of Blockchain, Centralized vs Decentralized systems. Types of Blockchain- Public, Private. Consortium. Blockchain Architecture- Basic block

structure, Merkle tree. Applications of Blockchain.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Installing and Configuring Wireshark i. Installation of Wireshark on the system ii. Basic configuration and setup for packet capturing
2	Capturing packets with Wireshark and working with captured packets and measure RTT, packet loss, throughput.
3	Basic Network Configuration using Cisco Packet Tracer and analysing lower layer protocols Ethernet frame format, ARP, TCP, UDP, ICMP.
4	Configuring different types of switches.
5	Implementing router and switch administrative configurations: Hostnames, Banners, setting passwords, encrypting passwords.
6	Implementing switch administrative configurations such as viewing, saving and erasing configurations.
7	Configuring VLANs in Cisco Packet Tracer.
8	Configuring Trunk Ports in Cisco Packet Tracer.
9	Configuring a Router for Routing Information Protocol (RIP) in Cisco Packet Tracer.
10	Configuring a Router for OSPF in Cisco Packet Tracer.
11	Create Wireless LAN in Packet Tracer.
12	To capture and analyze IEEE 802.11 wireless frames (Management, Control, Data frames) using Wireshark.

Text Books

1. J. F. Kurose and K. W. Ross, 'Computer Networking: A Top-Down Approach", Pearson, 6th edition, 2012.
2. L. L. Peterson and B. S. Davie, 'Computer Networks: A Systems Approach", Morgan Kaufmann, 5th edition, 2012.
3. N. F. Mir, "Computer and Communication Networks", Pearson Education, 2nd edition, 2015.
4. Bahga, Arshdeep, and Vijay Madisetti., "Internet of Things: A Hands-on Approach", Universities Press, 1st edition, 2015.
5. Bashir, Imran., "Mastering Blockchain: Inner Workings of Blockchain, from Cryptography and Decentralized Identities, to DeFi, NFTs and Web3", Packt Publishing, 4th edition, 2023.

Reference Books

1. B. A. Forouzan, "Data Communications and Networking", McGraw-Hill Education, 4th edition, 2007.
2. J. S. Beasley and P. Nilkaew, "A Practical Guide to Advanced Networking", Pearson, 3rd edition, 2012.
3. A. S. Tanenbaum and D. J. Wetherall," Computer Networks",Prentice Hall, 5th edition, 2010.
4. Antonopoulos, Andreas M., and David A. Harding., "Mastering Bitcoin: Programming the Open Blockchain", O'Reilly Media, 3rd edition, 2024
5. W. Stallings, "Data and Computer Communications", Prentice Hall, 7th edition, 2003.

Online Learning Materials

1. http://www.innos.in/downloads/CISCO%20-%20640-802-ccna.pdf Accessed on December 12, 2025
2. https://youtu.be/frUQMHXhnvs?si=kYtyA8S8f6thfTV2 Accessed on December 12, 2025
3. https://youtu.be/xwaqSZHK8eM?si=wlPtPLPPG21-hkHJ Accessed on December 12, 2025
4. https://users.cs.fiu.edu/~prabakar/cen5079/Common/textbooks/Mastering_Blockchain_2nd_Edition.pdf Accessed on December 12, 2025
5. https://www.youtube.com/watch?v=CaukSKg_sI0 Accessed on December 12, 2025
6. https://www.kwcsangli.in/uploads/Computer_Networking_A_Top-Down_Approach_Ross.pdf Accessed on December 12, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Advanced Computer Networks	Prof. Neminath Hubballi, Prof. Sameer G Kulkarni	IIT Indore, IIT Gandhi Nagar	https://onlinecourses.nptel.ac.in/noc23_cs35/preview
2	Introduction to Internet of Things	Prof. Sudip Misra	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc22_cs53/preview
3	Blockchain and its Applications	Prof. Sandip Chakraborty, Prof. Shamik Sura	IIT Kharagpur	https://nptel.ac.in/courses/106105235

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS105
Course Title: Object Oriented Analysis and Design using UML

Programme: B.Tech.	**L:**3 **T:** 0 **P:2**	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective-I		

Prerequisites (If any): CCS112

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply object-oriented modeling principles and UML notations to represent real-world systems.
2	Develop requirement models using use case diagrams, activity diagrams, and structured use case descriptions.
3	Analyze problem statements to identify classes, attributes, operations, relationships, and construct class/object/package diagrams.
4	Analyze dynamic system behavior using sequence, collaboration, activity, and state machine diagrams with events and concurrency.
5	Evaluate alternative architectural models and justify system design decisions using UML.
6	Design complete UML based solutions using advanced UML concepts, extensibility, and CASE tools for scalable systems.

Contents

Part-A

Unit-1 Introduction **8 (L) hrs**

Introduction about Modeling & Model, Principles of modeling, Object-oriented modeling concepts, Unified Modeling Language (UML) history and goals, UML Architecture and Versions, UML Building Blocks-Things, Relationships, Diagrams, Rules and Common Mechanisms (specification, adornments, extensibility).

Unit-2 Requirements Modeling **7(L) hrs**

Actors, Use Cases, and Scenarios, Use case diagrams -Notation and Semantics, Writing effective use case descriptions, Relationships-Include, Extend, Generalization, Activity diagrams for workflows, Use-case driven development, Traceability from requirements to design.

Unit-3 Structural Modeling **7(L) hrs**

Classes, Interfaces, Attributes, Operations, Relationships-Association, Aggregation, Composition, Generalization, Dependency, Class diagrams - Modeling Vocabulary of a System, Object Diagrams-Snapshots of Instances, Package Diagrams and Model organization, Modeling Design Classes vs Analysis Classes.

Part-B

Unit-4 Behavioral Modeling **8(L) hrs**

Interaction diagrams - Notations, Conditional messaging, Branching, Time constraints, Sequence diagram, Collaboration diagram and Activity diagrams. Signal events, call events, time events, and change events, modeling a family of signals, Modeling exceptions, Handling events in active and passive, State Machines, transitions, and activities, Modeling the lifetime of an object, Creating well-structured algorithms, Active objects, processes and threads, Modeling multiple flows of control, Modeling inter process communication, Building thread-safe abstractions.

Unit-5 Architectural Modeling **8(L) hrs**

Modeling a Client/Server System, Modeling a Fully Distributed System, Systems, subsystems, models, and views, Modeling the architecture of a system, Modeling systems of systems, Organizing the artifacts of development.

Unit-6 Advanced UML Concepts **7(L) hrs**

UML Extensibility- Stereotypes, Tagged Values, Constraints, UML Profiles, Modeling Concurrency and Threads, Modeling Large-Scale Systems, UML in Iterative & Agile development, Tool-based UML modeling.

Laboratory Work

In the laboratory work, students will implement programs related to the above topics using any open-source UML tools, such as StarUML, Modelio, ArgoUML, Umbrello etc.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	To analyze a real-world problem and design its basic UML model by identifying objects, classes, and relationships to justify the need for modeling.
2	To design a UML model by identifying its building blocks, relationships, and applying common mechanisms such as specification, adornments, and extensibility.
3	To design a complete use case model by identifying actors and use cases of a real-world system using a UML CASE tool.
4	To design detailed use cases and activity models representing workflows and relationships using UML.
5	To design an analysis class diagram by identifying classes, attributes, operations, and associations from use case descriptions.
6	To design system structure by converting analysis classes into design class, object, and package diagrams using UML.
7	To design a Sequence Diagram for the selected use case showing lifelines, message flow, and control logic, then transform it into a Communication Diagram representing the same interactions and object links.
8	To design a State Machine Diagram for a reactive object and compare it with an Activity Diagram to highlight differences in modeling states versus workflows.
9	To design a Component Diagram for a layered or client–server application showing components, their interfaces, and dependencies.
10	To design a Deployment Diagram mapping software artifact to hardware nodes to model client–server or distributed system architecture.
11	To apply UML stereotypes, tagged values, and constraints to an existing model and create a domain-specific UML Profile.
12	To model concurrency and threads in UML using a CASE tool to build a complete iterative UML model.
13	Any one project: i. Online Book Store System ii. University Course Management System iii. Online Banking System iv. Hospital Management System Any other project recommended by the instructor.

Text Books

1. Grady Booch, James Rumbaugh, Ivar Jacobson, “Unified Modeling Language User Guide”, Addison-Wesley Professional, 2nd Edition, 2005.
2. James Rumbaugh, “Object-Oriented Modeling and Design with UML”, Pearson Prentice Hall, 2nd Edition , 2005.
3. Roger S Pressman, “Software Engineering: A practitioner’s approach”, McGraw Hill Education, 7th Edition, 2010.

4. Mahesh. P. Matha, "Object Oriented Analysis and Design Using UML", PHI Publication, 3rd Edition, 2008.
5. Ian Sommerville, "Software Engineering", Pearson India, 10th Edition, 2018.
6. Rajesh Gupta, Paul Le Guernic, Sandeep Kumar Shukla, Jean-Pierre Talpin, "Formal Methods and Models for System Design: A System Level Perspective", Springer Science & Business Media, 1st Edition, 2004.

Reference Books

1. Ali Bahrami, "Object Oriented Systems Development", McGraw-Hill, 1st Edition, 1999.
2. Meilir Page-Jones, "Fundamentals of Object-Oriented Design in UML", 1st Edition, Addison-Wesley, 1st, Edition, 2000.
3. Donald W. Boyd, "Systems Analysis and Modeling: A Macro to Micro Approach with Multidisciplinary Applications", Academic Press , 1st Edition, 2001.

Online Learning Materials

1. https://www.tutorialspoint.com/object_oriented_analysis_design/ooad_uml_analysis_model.htm Accessed on 1-1-2026
2. https://www.tutorialspoint.com/object_oriented_analysis_design/ooad_object_oriented_design.htm Accessed on 1-1-2026
3. https://www.youtube.com/watch?v=WnMQ8HlmeXc Accessed on 1-1-2026
4. https://www.youtube.com/watch?v=FnaEQcQ8e9Y Accessed on 1-1-2026
5. https://www.youtube.com/watch?v=4emxjxonNRI Accessed on 1-1-2026
6. https://www.youtube.com/watch?v=el501qi0KtI Accessed on 1-1-2026
7. https://www.youtube.com/watch?v=sN2_CoB_kbw Accessed on 1-1-2026
8. https://www.youtube.com/watch?v=hVAxMFUG_R8 Accessed on 1-1-2026

Supplementary SWAYAM Course

S.No	Course Name	Instructor	Host Institute	URL
1	Object Oriented System Development Using UML, Java and Patterns	Prof. Rajib Mall	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc25_cs52/preview?utm_source=chatgpt.com
2	Object Oriented Analysis and Design	Prof. Partha Pratim Das	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc22_cs99/preview?utm_source=chatgpt.com

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS106
Course Title: Computer Graphics

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 5	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 25%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective - I		

Prerequisites: BSCS101

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply mathematical and programming concepts to solve computer graphics problems.
2	Evaluate scan conversion algorithms for generation of real-world objects.
3	Apply clipping algorithms, area-filling techniques, and color models in graphics applications.
4	Use advanced techniques for transformation and modification of 2D and 3D objects.
5	Implement algorithms for visualization, rendering and representation of graphical objects.
6	Analyze computer graphics concepts for the development of different graphical applications.

Contents

Part-A

Unit-1 Introduction **7(L)hrs**

Overview of computer graphics, Computer graphics applications, Different I/O devices with specialized graphics features, Elements of graphics. Graphic systems – Video display devices, Raster scan systems, Random scan systems. Video Controller.

Unit-2 Output Primitives **5(L)hrs**

Scan conversion basics, Algorithm for scan converting a point, Scan converting a line – Digital differential analyzer algorithm, Bresenham's line algorithm. Scan converting circle – Bresenham's circle drawing algorithm, Midpoint circle drawing algorithm. Scan converting ellipse – Midpoint

ellipse algorithm. Filling Techniques – Scan line polygon fill algorithm, Boundary-fill, Flood-fill. Anti-aliasing.

Unit-3 Two Dimensional Transformations 6(L)hrs

Geometric and coordinate transformations. Geometric transformations – Scaling, Rotation, Translation, Reflection, Shear. Matrix representations, Homogeneous coordinates, Composite transformations.

Part-B

Unit-4 Two Dimensional Viewing and Clipping 7(L) hrs

The viewing pipeline, Window-to-viewport transformation, Point clipping, Line clipping algorithms – Cohen-Sutherland, Liang-Barsky, Nicholl-Lee-Nicholl. Polygon clipping algorithms – Sutherland-Hodgeman, Weiler-Atherton. Curve and text clipping.

Unit-5 Three Dimensional Transformations and Viewing 7(L) hrs

3D geometric transformations – Scaling, Rotation, Translation, Reflection, Shear. Composite transformations, 3D viewing, Viewing pipeline, Parallel projections, perspective projections, classifications of projections

Unit-6 Visible Surface Detection 7(L) hrs

Classification of visible surface detection algorithms. Techniques for efficient visible surface algorithms – Back face detection, Depth-buffer method, A-buffer method, Scan line method, Depth sorting method, BSP tree Method, Area-subdivision method, Octree Methods, Ray casting method.

Unit-7 Surface Rendering and Color Models 6(L) hrs

Light sources, Surface lighting effects, Illumination models, Polygon rendering methods – Constant-intensity shading, Gouraud shading, Phong shading, Fast Phong shading, Color Models – Red Green Blue (RGB), Luminance In-phase Quadrature (YIQ), Cyan Magenta Yellow (CMY), Hue Saturation Value (HSV).

Laboratory Work

In the laboratory work, students should implement these concepts using any programming language like C++/Java etc.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Write a program for creating a simple two-dimensional shape of any object using lines, circle, etc.
2.	Implement the DDA algorithm for drawing line (programmer is expected to shift the origin to the center of the screen and divide the screen into required quadrants).
3.	Write a program to input the line coordinates from the user to generate a line using Bresenham‘s Algorithm.
4.	Write a program to generate a complete moving wheel using Midpoint circle drawing algorithm and DDA line drawing algorithm.
5.	Write a program to draw an ellipse using the Midpoint ellipse generation algorithm for both the regions.
6.	Write a program to draw any 2-D object and perform the transformations on it according to the input parameters from the user, namely: Translation, Rotation and Scaling.
7.	Write a program to Draw a color cube and spin it using transformation matrices.
8.	Write a program to rotate a triangle about any one of its end coordinates.
9.	Write a program to perform clipping on a line against the clip window using any line clipping algorithm. The output must be twofold showing the before clipping and after clipping images.
10.	Write a program to implement the Sutherland-Hodgeman Polygon Clipping algorithm for clipping any polygon.
11.	Write program to draw a house like figure and perform the following operations. a) Scaling about the origin followed by translation. b) Scaling with reference to an arbitrary point.
12.	Write a program to draw a 4×4 chessboard rotated 45˚ with the horizontal axis. Use Bresenham‘s algorithm to draw all the lines. Use seed fill algorithm to fill black squares of the rotated chessboard.
13.	Write a program to remove hidden surfaces using Scan line method.
14.	Write a program to implement Back Face Detection algorithm to remove hidden surfaces.
15.	Write a program to implement Gouraud and Phong shading on the surface of an object.

Text Books

1. D. Hearn and M.P. Baker, Computer Graphics, Second Edition, PHI/Pearson Education.

2. Zhigang Xiang, Roy Plastock, Theory and Problems of Computer Graphics, Second Edition, Tata McGraw-Hill.
3. C. Foley, Van Dam, Feiner and Hughes, Computer Graphics Principles & Practice, Second Edition, Pearson Education.
4. Amarendra N. Sinha, Arun D. Udai, Computer Graphics, First Edition, Tata McGraw-Hill.
5. N. Krishnamurthy, Introduction to Computer Graphics, First Edition, Tata McGraw-Hill.

Reference Books

1. Malay, K. Pakhira ,Computer Graphics, Multimedia and Animation, Second Edition, PHI.
2. Rogers, Adams, Mathematics Elements for Computer Graphics, Second Edition, Tata Mc-Graw Hill.

Online Learning Materials

1. https://www.youtube.com/watch?v=fwzYuhduME4 Accessed on 1-1-2026
2. https://www.coursera.org/learn/interactive-computer-graphics Accessed on 1-1-2026
3. https://www.tutorialspoint.com/computer_graphics Accessed on 1-1-2026

Supplementary SWAYAM Course

S.No	Course Name	Instructor	Host Institute	URL
1	Computer Graphics	Prof. Sukhendu Das	IIT Madras	http://nptel.ac.in/courses/106106090
2	Computer Graphics	Prof. Samit Bhattacharya	IIT Guwahati	nptel.ac.in/courses/106103224

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS107
Course Title: Natural Language Processing

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L)+30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Internal Assessment (CIA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: Nil		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective-II		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply the knowledge of mathematics and engineering to understand the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
2	Implement and evaluate part-of-speech tagging and parsing techniques using probabilistic models and context-free grammars.
3	Demonstrate the concepts of morphology, syntactic analysis, semantic interpretation and pragmatics of the language, with use of different approaches.
4	Examine lexical semantics and word representation techniques, including embeddings, to measure semantic similarity and meaning in text.
5	Apply speech processing concepts and feature extraction techniques for analyzing speech signals in NLP systems.
6	Design and assess end-to-end NLP applications such as information extraction, question answering, machine translation, coreference resolution, dialogue systems, and speech-based language technologies.

Contents

Part-A

Unit-1 Introduction **6(L)hrs**

Introduction to natural language and speech processing, Steps for processing natural languages, Issues and challenges for processing of natural languages, Applications of natural language processing.

Unit-2 Morphology and N-gram Language Models **8(L)hrs**

Words, Morphemes, Inflectional and Derivational morphology, Unicode, Subword Tokenization, Byte-Pair Encoding, Corpora, Regular Expressions, Rule-based tokenization, Minimum Edit

Distance, N-gram Language Models, Evaluating Language Models – Training and Test Sets, Perplexity, Overfitting the training set, Smoothing, Interpolation, and Backoff, Entropy.

Unit-3 Part-of-Speech Tagging and Parsing **9(L)hrs**

Word Classes, Part-of-Speech Tagging, Named Entities and it's Tagging, HMM Part-of-Speech Tagging, Context-Free Grammars, Treebanks, Ambiguity, CKY Parsing, Dependency Relations, Information Extraction – Relation Extraction, Extracting Events, Representing Time, Representing Aspect.

Part-B

Unit-4 Lexical Semantics **8(L)hrs**

Lexical Semantics, Vector Semantics, Embeddings – Count-based embeddings, Cosine for measuring similarity, Visualizing Embeddings, Bias and Embeddings. Semantic Role Labeling – Semantic Roles, Diathesis Alternations, Problems with Thematic Roles, The Proposition Bank, Semantic Role Labeling, Selectional Restrictions

Unit-5 Phonetics and Speech Feature Extraction **6(L)hrs**

Speech Sounds and Phonetic Transcription, Prosody, Acoustic Phonetics and Signals, Feature Extraction for Speech Recognition – Log Mel Spectrum, Mel Frequency Cepstral Coefficients.

Unit-6 Coreference Resolution and Discourse Coherence **8(L)hrs**

Coreference Resolution – Coreference Phenomena, Coreference Tasks and Datasets, Mention Detection, Architectures for Coreference Algorithms, Classifiers using hand-built features, Entity Linking. Discourse Coherence – Coherence Relations, Discourse Structure Parsing, Centering and Entity-Based Coherence, local coherence, Global Coherence.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Use Naïve Bayes method to classify positive or negative sentiment in tweets.
2.	Use the auto-correct algorithm to implement a simple spell checker, using unigram frequency to sort options at similar edit distance.
3.	Implement a simple auto-correct algorithm using minimum edit distance and dynamic programming.
4.	Implement the auto-complete algorithm using an N-gram model.
5.	Apply the Viterbi algorithm for Part-of-Speech tagging.
6.	Implement HMM for Part-of-Speech tagging.

7.	Implement the CKY Parser using Dynamic programming.
8.	To measure the similarity between two sentences using cosine similarity.
9.	Extract the Mel Frequency Cepstral Coefficients (MFCC) from speech signal for feature representation.
10.	Implement coreference resolution algorithm for identifying and linking referring expressions in a given text.

Text Books

1. D. Jurafsky and J. H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 3rd Edition.
2. J. Allen, Natural Language Understanding, Second Edition, Addison Wesley.
3. Andrew Radford, Martin Atkinson, David Britain, Harald Clahsen, Andrew Spencer - Linguistics, An Introduction", Second Edition, Cambridge University Press.

Reference Books

1. T. Siddiqui and U.S. Tiwary, - Natural Language Processing and Information Retrieval‖, First Edition, Oxford University Press.
2. J. Handke, - The Structure of the Lexicon: Human Versus Machine (Natural Language Processing), First Edition, Mouton de Gruyter.
3. Bharati, V. Chaitanya and R. Sangal, - Natural Language Processing: A Paninian Perspective, Third Edition, Prentice Hall of India.

Online Leaning Material

1. https://www.ibm.com/think/topics/natural-language-processing Accessed on: 10-01-2026
2. https://www.oracle.com/in/artificial-intelligence/natural-language-processing/#what-is-nlp

Accessed on: 10-01-2026

Supplementary SWAYAM Courses

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Natural Language Processing	Prof. Pawan Goyal	IIT Kharagpur	https://nptel.ac.in/courses/106105158
2	Natural Language Processing	Prof. Pushpak Bhattacharyya	IIT Bombay	https://nptel.ac.in/courses/106101007

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS108
Course Title: Big Data

Programme: B.Tech.	**L:** 3 **T**:0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – II		

Prerequisites (if any): CCS104

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply Big Data concepts, architectures, and ecosystem components to select appropriate tools and technologies for handling large-scale, heterogeneous data sets.
2	Implement distributed data storage and processing using Hadoop HDFS and MapReduce.
3	Analyze data processing needs to choose appropriate Hadoop ecosystem tools such as Hive, Pig, and Spark.
4	Analyze NoSQL database models to identify suitable solutions for Big Data storage and retrieval.
5	Evaluate Big Data analytics frameworks for effective large-scale data analysis and decision-making.
6	Design and develop an end-to-end Big Data application using Hadoop, Spark, and NoSQL tools.

Contents

Part-A

Unit-1 Introduction to Big Data **5(L)hrs**

Big Data Overview - Definitions, characteristics of Big Data (5Vs), Evolution of Big Data, Big Data in industry. Data Structures, Traditional BI vs Big Data Analytics, Analytical Architecture, Batch Processing vs Near Real-Time Processing, Batch Processing- Concept and use cases (e.g., payroll processing, sales report generation, historical log analysis using Hadoop MapReduce). Near Real-Time Processing- Concept and use cases (e.g., fraud detection, social media trend analysis, sensor data monitoring using Spark Streaming). Data at Rest vs Data in Motion, Examples of Big Data Analytics tools - Hive, HBase, Spark, Kafka, MongoDB. Roles in Big Data ecosystem – Data Engineer, Data Analyst, Data Scientist, Architect.

Unit-2 Hadoop Ecosystem and HDFS **8(L) hrs**

Introduction to Hadoop - RDBMS vs Hadoop comparison, History of Hadoop, Google File System, Hadoop Components - HDFS, YARN, MapReduce. HDFS Design - NameNode, DataNode, Secondary NameNode, metadata, replication, block structure. Data Flow in Hadoop, Hadoop I/O – Serialization, Cluster Setup & Configuration- Local, pseudo-distributed, fully distributed mode. Managing Hadoop-Configuration XML files, web-based monitoring tools, Hadoop as batch-oriented processing framework.

Unit-3 Map Reduce Programming **7(L)hrs**

Introduction to MapReduce - Programming model, anatomy of a MapReduce job. Building Blocks - Mapper, Reducer, Combiner, Partitioner, Input Format & Output Format (Text, Key Value, Sequence files), Advanced Features - Counters, Custom Writables, Steps to Read Write in HDFS. Shuffle & Sort Mechanism - Failures, fault tolerance. WordCount with predefined Mapper & Reducer, Local Testing vs Cluster Execution. Comparison- MapReduce vs Spark execution model (conceptual overview).

Unit-4 Working with Hadoop Tools (Hive, Pig) **6(L)hrs**

Hive Introduction – Hive Overview, Architecture, Data types, Hive Metastore. Hive Query Language (HQL) - DDL, DML, Joins, Views, Partitioning, Bucketing. User-Defined Functions (UDFs), Pig Introduction - What is Pig, Pig Latin Language, Data types, operators. Pig Execution Modes - Local, MapReduce mode. Pig use cases vs Hive use cases. Use cases and comparative analysis of Hive, Pig, and Spark for batch analytics.

Part-B

Unit-5 NoSQL Databases **7(L)hrs**

MongoDB – Introduction to MongoDB, MongoDB vs RDBMS, MongoDB data types, CRUD operations, HBase - HBase vs RDBMS, Column families, HFiles. HBase Architecture - HMaster, Region servers, Zookeeper. HBase operations - Creating tables, inserting data, scanning, MapReduce integration. Cassandra - CAP theorem, eventual consistency. Data model - Keyspace, column families.

Unit-6 Spark and Big Data Analytics **7(L) hrs**

Spark Essentials - Spark Ecosystem, RDDs - Transformations & Actions, DAG, Spark SQL - DataFrames, Datasets, SQL Operations, Window Functions. Spark MLlib (Machine Learning) - Introduction to Big Data Analytics, Regression (Linear Regression), Classification (Logistic Regression), Clustering (K-Means). Introduction to Spark Streaming for near real-time analytics.

Unit-7 Business Analytics and Text Analytics **5(L) hrs**

Categorization of analytics methods and models – Descriptive analytics, Predictive analytics, Perspective analytics. Introduction to Text Mining - Tokenization, Stemming, Stop-words. Case Studies using Spark-based analytics pipelines.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Installation & Configuration of Hadoop (Single Node Cluster)
2	Working with HDFS - Create directories in HDFS, Upload, download, delete files, Understand block replication, Check file storage using hdfs dfsadmin commands
3	Use web-based tools to monitor your Hadoop cluster and perform tasks.
4	MapReduce – WordCount Program - Write WordCount in Java/Python, Compile, package as JAR, Run job on Hadoop cluster, Observe job summary and counters
5	Implementation of Advanced MapReduce Algorithms Including Sorting, Partitioning, and Combining
6	Introduction to Apache Spark- Environment Setup and basic RDD operations using PySpark / Spark shell.
7	Spark Transformations and Actions – Data processing using RDDs and Spark SQL (DataFrames and SQL Queries)
8	Creation and Management of Databases and Tables in Hive with HQL-Based Data Analysis
9	Performing CRUD Operations and Aggregation Queries on MongoDB NoSQL Database
10	Creation, Manipulation, and Querying of Tables in HBase/Cassandra Distributed NoSQL Systems
11	Spark MLlib Experiments – Implementation of Linear Regression / Logistic Regression / K-Means clustering
12	Execution of any of the following End-to-End Mini Project Integrating Hadoop, Spark, and NoSQL Technologies: i. Movie Recommendation System ii. Fraud Detection using MLlib iii. Web Log Analysis using Spark iv. HBase-based Customer Profile Store v. Real-time Stream Analytics (Spark Streaming optional)

Text Books

1. Tom White, "Hadoop: The Definitive Guide", O'Reilly Media, Fourth Edition, 2015.

2. Seema Acharya, Subhasini Chellappan, "Big Data Analytics", Wiley, First Edition, 2015.
3. Parag Kulkarni, Sarang Joshi,S.Brown, "Big Data Analytics", PHI Learning Pvt. Ltd, 2nd Edition, 2016.
4. Paul C. Zikopoulos, Chris Eaten, Dirk Deroos, "Understanding Big Data", McGraw Hill, 1st Edition, 2012.
5. Chuck Lam, "Hadoop in Action", Dreamtech Press, Reprint edition, 2016.

Reference Books

1. EMC Education Services, "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", EMC2, Wiley Publications, First Edition,2015.
2. Jeffrey D. Camm, "Essentials of Business Analytics", CENGAGE Learning, First Edition, 2017.
3. Eric Siegel, Thomas H. Devanport, "Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die", Wiley Publications, First Edition,, 2016.

Online Learning Material

1. Lecture Notes on Big Data and Business Analytics by Ms. G. Sulakshana and Ms. G. Srilekha https://www.iare.ac.in/sites/default/files/NEW%20LECHURE%20NOTES.pdf
   Accessed on 2 Jan 2026
2. Online Book —Big Data- Principles and Paradigms by Rajkumar Buyya http://dhoto.lecturer.pens.ac.id/lecture_notes/internet_of_things/Big%20Data%20Principles%20and%20Paradigms.pdf
   Accessed on 2 Jan 2026

Supplementary SWAYAM Courses

S. No.	Course Name	Instructor	Host Institute	URL
1.	Big Data Computing	Prof. Rajiv Misra	IIT Patna	https://onlinecourses.nptel.ac.in/noc25_cs131/preview
2.	Big Data Analytics for Smart Grid	Dr. Ritula Thakur	NITTR Chandigarh	https://onlinecourses.swayam2.ac.in/arp19_ap60/preview
3.	Introduction to Data Analytics	Prof. Nandan Sudarsanam, Prof. Balaraman Ravindran	IIT Madras	https://nptel.ac.in/courses/110106072

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS109
Course Title: Information Security Management

Programme: B.Tech.	**L:**3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30 (P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective-II		

Prerequisites (if any): CCS106, CCS111

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand the principles and components of information security management.
2	Analyze security threats, attacks, and system vulnerabilities.
3	Apply legal and ethical requirements in information security.
4	Perform risk assessment and select appropriate risk controls.
5	Implement security technologies and cryptographic solutions.
6	Manage information security operations, personnel, and maintenance.

Contents

Part-A

Unit-1 Introduction and Significance **9(L) hrs**

The History of Information Security, Components of an Information System, Balancing Information Security and Access, Approaches to Information Security Implementation. Threats and Attacks, Compromises to Intellectual Property, Deviation in QoS, Human Error, Information Extortion, Sabotage or Vandalism, Software Attacks, Technical Hardware/Software Failure, Technological Obsolescence, and Theft.

Unit -2 Legal and Ethical Issues **6(L) hrs**

Law and Ethics in Information Security, International Laws and Legal Bodies, Ethics and Information Security, Codes of Ethics of Professional Organizations.

Unit-3 Planning and Risk Management **8(L)hrs**

Information Security Planning and Governance, Information Security Policy, Standards, and Practices, The Information Security Blueprint, Security Education, Training, and Awareness. An

Overview of Risk Management, Risk Identification, Risk Assessment, Risk Control, Quantitative Versus Qualitative Risk Management Practices, and Recommended Risk Control Practices.

Part-B

Unit-4 Cryptology and Security Technology **7(L)hrs**

Foundations of Cryptology, Access Control, Firewalls, Protecting Remote Connections**,** Intrusion Detection and Prevention Systems, Honeypots, Honeynets, and Padded Cell Systems, Scanning and Analysis Tools.

Unit -5 Implementing Information Security **8(L)hrs**

Information Security Project Management, Technical Aspects of Implementation, Nontechnical Aspects of Implementation, Information Systems Security Certification and Accreditation.

Unit- 6 Security Personnel and Maintenance **7(L)hrs**

Positioning and Staffing the Security Function, Credentials for Information Security Professionals, Employment Policies and Practices, Security Management Maintenance Models, and Digital Forensics.

Laboratory Work

In the laboratory work, students will implement programs related to the above topics using any open-source tools : Kali Linux, Wireshark, Snort, OpenSSL

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Study of information security components, assets, threats, and vulnerabilities.
2.	Analysis of cyber threats and attacks using case studies
3.	Preparation of an information security policy with legal and ethical considerations
4.	Risk identification and qualitative risk assessment
5.	Quantitative risk analysis (SLE, ALE calculations)
6.	Implementation of access control mechanisms
7.	Firewall configuration and traffic filtering
8.	Vulnerability scanning and analysis using security tools

9.	Implement and document an information security plan incorporating project management, technical controls and non-technical controls.
10.	Digital forensics: evidence collection, analysis, and reporting

Text Books

1. Michael E. Whitman and Herbert J. Mattord, "Principles of Information Security",, Cengage Learning, 7th Edition, 2021.
2. Mark Stamp, "Information Security: Principles and Practice",, Wiley, 3rd Edition, 2019.

Reference Books

1. William Stallings and Lawrie Brown, "Computer Security: Principles and Practice", Pearson, 5th Edition, 2020.
2. Harold F. Tipton and Micki Krause, "Information Security Management Handbook", CRC Press, 7th Edition, 2018.
3. Charles P. Pfleeger and Shari Lawrence Pfleeger, "Security in Computing", Prentice Hall, 5th Edition, 2015.

Online Learning Materials

1. https://nptel.ac.in/courses/106106129 Accessed on December 16, 2025
2. https://onlinecourses.nptel.ac.in/noc25_ee54/preview Accessed on December 16, 2025
3. https://onlinecourses.nptel.ac.in/noc24_cs121/preview Accessed on December 16, 2025

Supplementary SWAYAM Courses

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Information Security	Dr. Maninder Singh	TIET (Deemed to be University), Patiala	https://onlinecourses.swayam2.ac.in/cec24_cs15/preview
2	Information Security	Dr. J. Kokila	IIIT, Tiruchirappalli	https://onlinecourses.swayam2.ac.in/ntr25_ed134/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS110
Course Title: Network Security and Cryptography

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 40%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – II		

Prerequisites (if any): CCS106

Additional Material Allowed in ESE: Scientific Calculator

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Identify and analyze network security threats and design appropriate security models and mechanisms.
2	Apply number theory and algebraic concepts to understand cryptographic algorithms.
3	Implement and evaluate classical and modern symmetric-key cryptographic algorithms and hash functions.
4	Understand and analyze public-key cryptographic techniques including RSA and key exchange mechanisms.
5	Apply authentication, key management, and Internet security protocols to provide secure communication.

Contents

Part-A

Unit-1 Introduction to Security **4 (L) hrs**

Essentials of network security, Architecture, Security goals. Cryptographic attacks - cryptanalytic, non-cryptanalytic attacks, active attack and passive attack, security Services and security mechanism, Fundamental Security design principles, Network security model, standards.

Unit-2 Number Theory **6 (L) hrs**

Integer Arithmetic, Modular Inverse, Euclidean Algorithm, Extended Euclidean Algorithm, Modular Arithmetic, Matrices, Linear Congruence, Prime numbers, Euler's Phi-Function, Euler's Theorem, Fermat's, Factorization, Chinese Remainder Theorem and its applications.

Unit-3 Traditional Symmetric-Key Cipher **10 (L) hrs**

Concept of Symmetric-Key Ciphers: Encryption, Decryption, Plaintext, Cipher text and Symmetric cipher model, Cryptanalysis attacks. Categories of Tradition ciphers - Substitution cipher and transposition ciphers. Substitution techniques - Mono-alphabetic ciphers (additive, Caesar, Multiplicative, affine), polyalphabetic cipher (autokey, Vigenère, playfair, Hill Cipher), Transposition techniques (keyless, keyed, combined approaches).

Part-B

Unit-4 Modern Symmetric-key Ciphers **8 (L) hrs**

Modern Block cipher, components of block cipher, two classes of product cipher, Feistal structure. Data Encryption Standard (DES) - concept, structure, DES analysis and properties. Modern stream ciphers, Advanced Encryption Standard (AES), comparison between DES and AES, Use of Modern Block Cipher: modes of operation, Use of Stream ciphers – RC4.

Unit-5 Asymmetric-Key Cryptography **7 (L) hrs**

Symmetric – Key vs Asymmetric-key cryptosystems, RSA cryptosystem: Principle, key generation and procedure, Attacks of RSA, Diffie Hellman Key Exchange.

Unit-6 Message Integrity and Message Authentication **4(L)hrs**

Cryptographic Hash Function Criteria, Secure Hash Algorithm (SHA-512), Message Authentication code - Authentication requirements, MAC, Message Authentication (MD5). Digital Signature - services, attacks on digital signature, RSA Digital signature scheme.

Unit-7 Internet Security Protocols **6 (L) hrs**

General structure of Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET). IP Security – Overview, IP security architecture modes. security protocols - Authentication header (AH) and Encapsulation security payload (ESP). Intrusion Detection Systems (IDS) - Types of IDS Technologies, False Positives and Negatives, Intrusion Detection Techniques, Firewalls and its types.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	Analyze network traffic to capture and study different protocol packets using i. Wireshark ii. Snort

	iii. tcpdump
2.	Implement Euclidean Algorithm for finding GCD in Cryptographic Applications.
3.	Implement the Vigenère Cipher for secure encryption and decryption of text using a keyword in C++.
4.	Implement the Caesar Cipher to encrypt and decrypt a given plaintext using a fixed shift key in C++.
5.	Implementation and Performance evaluation of DES cryptographic algorithms using C++ .
6.	Implementation and Performance evaluation of RSA cryptographic algorithms using C++ .
7.	Implement the Diffie-Hellman Key Exchange mechanism in C++.
8.	Generate the message digest of a given text using the MD5 algorithm in C++.
9.	Generate the message digest of a given text using the SHA hash algorithm in C++.
10.	Perform port scanning of a host system using NMAP to identify open and closed ports.
11.	Study Intrusion Detection Systems and Honey pots.

Text Books

1. William Stallings, “Cryptography and Network Security Principles and Practice”, Pearson Education, 6th Edition, 2013.
2. Behrouz A. Forouzan, “Cryptography & Network Security”, McGraw-Hill Education, 3rd Edition, 2015.
3. Atul Kahate, “Cryptography & Network Security”, Tata Mc Graw Hill, 3rd Edition, 2013

Reference Books:

1. Wenbo Mao, Modern Cryptography: Theory and Practice, Hewlett-Packard Company, 1st Edition, 2003.
2. William Stallings, "Network Security Essentials, Applications and Standards ", Pearson Education,6th Edition, 2017.
3. Trappe & Washington, “Introduction to Cryptography with Coding Theory”, Pearson, 3rd Edition, 2021.

Online Learning Materials

1. Modern Cryptography by P. Rogaway https://web.cs.ucdavis.edu/~rogaway/classes/227/winter00/ , Accessed on 05-01-2026
2. A Graduate Course in Applied Cryptography by Dan Boneh, Accessed on 05-01-2026

3. Lecture Notes on Cryptography by S. Goldwasser and M. Bellare
http://cseweb.ucsd.edu/~mihir/papers/gb.pdf , Accessed on 05-01-2026

Supplementary SWAYAM Course

Sr. No.	Instructor	Host Institute	URL
1.	Prof. Sourav Mukhopadhyay	IIT Kharagpur	https://nptel.ac.in/courses/106105162
2.	Prof. Gaurav S. Kasbekar	IIT Bombay	https://onlinecourses.nptel.ac.in/noc25_ee54/preview
3.	Dr. Debdeep Mukhopadhyay	IIT Kharagpur	https://nptel.ac.in/courses/106105031

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS111

Course Title: Agile Software Development and DevOps

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 04
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 10%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – II		

Prerequisites (if any): CCS112

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Apply agile principles to build software iteratively and adapt to changing requirements.
2.	Analyze the core practices behind specific agile methodologies.
3.	Use version control concepts and tools to manage source code in software development.
4.	Apply the concepts of continuous integration and continuous delivery for automation.
5.	Elaborate security vulnerabilities and policies for secure development practices.
6.	Make use of tools to collaborate in cross functional teams and enhance communication.

Contents

Part-A

Unit-1 Introduction **5(L) hrs**

Need for agile software development, agile context - Agile Manifesto, Agile principles, values, stakeholders, Artifacts, and challenges. Business benefits of software agility.

Unit-2 Agile Methodologies **11(L) hrs**

Agile Methodologies - Extreme Programming, Scrum, Lean Software Development (LSD), Kanban, Adaptive software development (ASD). Test Driven Development, Agile Requirement and Design - User Stories, Story Boards, UI Sketching and Story Cards.

Unit-3 Version Control System **6(L) hrs**

Need, Version control system and its types, version control concepts - Repositories, commit, branch, merge, and clone. Git features, Git three stage models, basic git commands. Git and distributed version control.

Part-B

Unit-4 Continuous Integration and Continuous Delivery (CI/CD) 12(L) hrs

Significance of CI/CD, Benefits. Pipeline Stages - Static Code Analysis, Automated unit testing and code coverage, Build Automation, Artifact repository, Testing, Release management and deployment. Measurement and Optimization - metrics. Introduction to containerization, Need, benefits of containerization, Containerization using Docker.

Unit-5 Security and Monitoring in CI/CD 6(L) hrs

Importance, common security vulnerabilities, integration into development lifecycle. Best Practices - code scanning and static analysis, dependency scanning for third party libraries, Dynamic Application security Testing, secrets management and secure configuration.

Unit-6 DevOps Culture and Collaboration 5(L) hrs

Significance of DevOps culture. Elements of DevOps culture, the relationship between culture and technology, key cultural principles - collaboration, trust, transparency, and continuous improvement, collaboration and communication in cross functional teams, Best practices for fostering a collaborative DevOps. DevOps tools for collaboration.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

In the laboratory work, students will implement practical related to above topics using any open-source tool(s).

Experiment No.	Experiment Title
1.	Case studies on Agile Development, DevOps life cycle & its stages.
2.	Demonstrate Git installation with account creation.
3.	Demonstrate create, use, clone and fork operations in github repository.
4.	Demonstrate various operations with branches (create, delete, Merge).
5.	Learn initial setup and configuration of Jenkins server.
6.	Create Job and manage using Jenkins server.
7.	Experiment plug-in management with Jenkins server.
8.	Demonstrate user role creation and management using Jenkins server
9.	Demonstrate integration of Git with Jenkins server.
10.	Demonstrate automated deployments using Jenkins server.
11.	Demonstrate Build and delivery pipelines using Jenkins server.

12.	Demonstrate Job Parameterization using Jenkins server.
13.	Demonstrate Command line executions using Jenkins server.
14.	Demonstrate node management in Jenkins server.
15.	Create a Repository on Docker Hub, build an image and push the image to Docker Hub.
16.	Practical with any one Cloud environment (AWS/Google/Azure) for DevOps.

Text Books

1. Ken Schawber, Mike Beedle, "Agile Software Development with Scrum", 1st Edition, Pearson, 2001.
2. Robert C. Martin, "Agile Software Development, Principles, Patterns and Practices", 1st International Edition, Prentice Hall, 2002.
3. Gene Kim, Jez Humble, Patrick Debois, John Willis, "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations", IT revolution Press 1st edition, 2016.

Reference Books

1. Deepak Gaikwad, Viral Thakkar, "DevOps Tools from Practitioner's Viewpoint" 1st Edition, Wiley publications, 2019.
2. Gene Kim, "The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win", IT Revolution Press, 2013.

Online Learning Material

1. https://ebooks.karbust.me/Technology/The%20DevOps%20Handbook%20How%20to%20Create%20World-Class%20Agility,%20Reliability,%20and%20Security%20in%20Technology%20Organizations%20by%20Gene%20Kim,%20Jez%20Humble,%20Patrick%20Debois,%20John%20Willis.pdf Accessed on 5-1-2026
2. https://learning.dell.com/content/dam/dell-emc/documents/en-us/2023KS_Narasimhan-A_Beginner's_Handbook_to_DevOps.pdf Accessed on 5-1-2026
3. DevOps Engineer Masters Program https://www.simplilearn.com/devops-engineer-masters-program-certification-training Accessed on 5-1-2026
4. Advanced Certificate Program in DevOps https://www.iiitb.ac.in/advanced-certificate-programme-in-devops Accessed on 5-1-2026
5. https://www.youtube.com/watch?v=WjwEh15M5Rw Accessed on 5-1-2026
6. https://www.youtube.com/watch?v=PvAotetOv20 Accessed on 5-1-2026

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Software Engineering	Prof. Rajib Mall	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc20_cs68/preview
2	Software Project Management	Prof. Rajib Mall, Prof. Durga Prasad Mohapatra	IIT Kharagpur, NIT Rourkela	https://onlinecourses.nptel.ac.in/noc22_cs107/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS112
Course Title: Digital Image Processing

Programme: B.Tech.	**L:** 3 **T**:0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 50%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – II		

Prerequisites: BSCS101

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Demonstrate understanding of fundamental principles of digital image processing.
2	Apply intensity transformation and filtering techniques to enhance digital images.
3	Analyze morphological image processing operations and algorithms to solve image analysis problems.
4	Implement lossless and lossy image compression techniques.
5	Implement appropriate image segmentation techniques for real-world image analysis applications.

Contents

Part-A

Unit-1 Digital Image Fundamentals **5 (L) hrs**

Image perception – light and human visual system; concepts of luminance, brightness, and contrast. Digital Image Representation Image sampling and Quantization. Basic relationships between pixels. Types of digital images - binary, grayscale, and color images. Color representation techniques including color models, pseudo-color image processing and full-color image processing.

Unit-2 Intensity Transformations and Filtering **12 (L) hrs**

Image Enhancement: Enhancement by point processing, Sample intensity transformation, Histogram processing, Image subtraction, Image averaging. Spatial filtering - Smoothing Spatial filters, Sharpening Spatial filters. Frequency domain - Fourier Transform, Low-Pass, High-Pass, Laplacian, Homomorphic filtering.

Unit-3 Morphological Image Processing **6 (L) hrs**

Morphological Operations - Erosion, Dilation, Opening & Closing, Hit or Miss Transforms, Basic Morphological Algorithms and Reconstruction.

Part-B

Unit-4 Image Compression **10 (L) hrs**

Coding Redundancy, Interpixel Redundancy, Fidelity Criteria, Image Compression Models, Lossless Compression Techniques - Huffman Coding, Arithmetic Coding, LZW Coding, Run-Length Coding. Lossy compression techniques – Predictive Coding, Vector Quantization, JPEG Compression.

Unit-5 Image Segmentation **12 (L)hrs**

Discontinuities detection – point, line and edge detection, Thresholding, Region-based segmentation – region growing, region splitting and merging, Region Segmentation using Clustering, Superpixels and Graph Cuts, Segmentation using Morphological Watersheds.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Illustrate the use of Image Processing Toolbox on MATLAB/PYTHON to design and implement image processing applications like image enhancement, image compression, image segmentation

Experiment No.	Experiment Title
1	Read and display digital images and Perform image sampling and quantization.
2	Convert images between binary, grayscale, and color formats and analyse image brightness, contrast, and basic pixel relationships.
3	Implement point processing techniques (contrast stretching, negative, log transform).
4	Perform histogram equalization and histogram matching.
5	Apply spatial filters: smoothing (mean, Gaussian) and sharpening (Laplacian, unsharp masking). d)
6	Implement frequency domain filtering using Fourier Transform (LPF, HPF, homomorphic filtering).
7	Implement basic morphological operations: erosion, dilation, opening and closing on binary images for noise removal, shape sharpening and smoothing.
8	Detect specific patterns or shapes in binary images using Hit or Miss transforms.
9	Identify different types of redundancies present in digital images.
10	implement various coding techniques for image data.

11	To compare various image coding techniques and evaluate image quality after compression using fidelity metrics.
12	Perform edge detection using Sobel, Prewitt, and Canny operators.
13	Implement threshold-based segmentation and region growing.
14	Implement segmentation using clustering algorithms and superpixels.
15	Implement segmentation using graph-cuts and morphological watersheds.

Text Books

1. Gonzalez C. R., Woods E. R., "Digital Image Processing, Pearson Education", 4th edition, 2018.
2. Anil K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall, 1st Edition, 1989.

Reference Books

1. McAndrew A., "Introduction to Digital Image Processing with Matlab", Thomson Course Technology, 1st Edition, 2004.
2. Low A., "Introductory Computer Vision and Image Processing", McGraw-Hill, 1st Edition, 1991.

Online Learning Materials

1. https://www.coursera.org/learn/introduction-computer-vision-watson-opencv
   Accessed on Dec 27, 2025
2. https://www.cse.iitd.ac.in/~pkalra/col783/
   Accessed on Dec 27, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Computer Vision And Image Processing - Fundamentals And Applications	Prof. M. K. Bhuyan	IIT Guwahati	https://onlinecourses.nptel.ac.in/noc26_ee31/preview
2	Digital Image Processing	Prof. Prabir Kumar Biswas	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc25_ee126/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS113
Course Title: Soft Computing

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 60		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – III		

Prerequisites (if any): NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand the fundamental concepts, principles, and significance of soft computing in intelligent systems.
2	Analyze uncertainty, imprecision, and learning mechanisms in computational problem-solving.
3	Apply appropriate soft computing approaches to model and solve real-world engineering problems.
4	Evaluate the performance and suitability of different soft computing techniques for given applications.
5	Design computational solutions by integrating multiple soft computing paradigms.
6	Demonstrate the ability to implement and interpret results of soft computing-based systems.

Contents

Part-A

Unit-1 Introduction **6 (L) hrs**

Limitations of hard computing, Concept and need of soft computing, Characteristics of soft computing systems, soft computing vs. hard computing, Components of soft computing, Applications of soft computing in artificial intelligence, Role of soft computing in modern AI and Generative AI systems.

Unit-2 Fuzzy Set Theory and Fuzzy Logic **9 (L) hrs**

Fuzzy sets and crisp sets, Membership functions and their types, Fuzzy set operations, Linguistic variables, Fuzzy relations, Fuzzy rules and reasoning. Fuzzy inference systems – Mamdani and Sugeno models, Defuzzification techniques. Applications of fuzzy logic in decision-making systems.

Unit-3 Artificial Neural Networks (ANN) **9 (L) hrs**

Biological neuron vs. artificial neuron, ANN architecture. Learning Paradigms – supervised,

unsupervised and reinforcement learning, Single-layer perceptron, multi-layer perceptron, Backpropagation algorithm, Recurrent neural networks, Applications of ANN in pattern recognition and prediction.

Part-B

Unit -4 Genetic Algorithms **7 (L) hrs**

Introduction to evolutionary computation, Genetic Algorithms and biological inspiration, Chromosome representation and encoding schemes, Population initialization, Fitness function formulation, Selection mechanisms, Crossover and mutation operators, Elitism, Convergence behavior, Advantages and limitations of genetic algorithms, Applications of genetic algorithms in optimization and feature selection.

Unit -5 Swarm Intelligence Techniques **7 (L) hrs**

Introduction to swarm intelligence, Collective behavior and self-organization, Particle swarm-based optimization concepts. Ant-inspired optimization concepts, Comparative analysis of swarm-based techniques, Applications in routing, scheduling, and optimization.

Unit-6 Hybrid Soft Computing Systems and Applications **7 (L) hrs**

Need for hybridization in intelligent systems, Integration of soft computing paradigms, Hybrid learning and optimization models, multi-objective optimization concepts, Case studies from healthcare, finance, robotics, and decision support systems, Challenges and future directions of soft computing.

Laboratory Work

Students will implement the laboratory experiments using open-source soft computing tools such as Python and Jupyter Notebook.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Implementation of basic soft computing concepts and tools.
2	Design and analysis of fuzzy sets and membership functions.
3	Implementation of fuzzy rule-based system for decision making.
4	Design of fuzzy inference system and analysis of defuzzification methods.
5	Implementation of single-layer neural network for classification.
6	Implementation of multi-layer neural network for prediction or pattern recognition.
7	Implementation of genetic algorithm for optimization problem.

8	Application of genetic algorithm for feature selection or parameter optimization.
9	Implementation of swarm intelligence technique for optimization.
10	Design and implementation of hybrid soft computing model.

Text Books

1. Saroj Kaushik and Sunita Tiwari, "Soft Computing: Fundamentals, Techniques and Applications", McGraw Hill Education (India), First Edition, 2018.
2. S. Rajasekaran and G. A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications", PHI Learning Private Limited, New Delhi, Second Edition, 2017.
3. Amit Konar, "Soft Computing: Fundamentals and Applications, McGraw Hill Education", Third Edition, 2014.

Reference Books

1. J.S. R. Jang, C.T. Sun and E. Mizutani, "Neuro-Fuzzy and Soft Computing", Pearson Education, India, First Edition, 2015.
2. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic: Theory and Applications", Prentice Hall of India, New Delhi, First Edition, 2006.

Online Leaning Material

1. Scikit-learn Documentation (Neural Networks and Optimization) https://scikit-learn.org/stable/documentation.html Accessed on: 09-01-2026
2. NumPy and SciPy Documentation (Optimization and Numerical Computing) https://numpy.org/doc https://docs.scipy.org Accessed on: 09-01-2026
3. SpringerLink – Soft Computing and Intelligent Systems https://link.springer.com Accessed on: 09-01-2026

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Soft Computing Techniques	Prof. Pabitra Mitra	IIT Kharagpur	https://onlinecourses.swayam2.ac.in/nou25_cs43/preview
2	Artificial Intelligence: Knowledge Representation and Reasoning	Prof. Deepak Khemani	IIT Madras	https://onlinecourses.swayam2.ac.in/nou25_cs38/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS114
Course Title: Data Science

Programme: B.Tech.	**L:** 3 **T**:0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 20%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – III		

Prerequisites (if any): ESC105, CCS104

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply the data science approach to understand real-world problems and identify relevant data and variables.
2	Perform statistical computations to summarize data and draw meaningful conclusions.
3	Implement basic R programming techniques to organize and transform datasets from multiple sources.
4	Analyze data quality, structure, and representation issues, including ethical considerations in data handling.
5	Evaluate datasets using exploratory data analysis to identify patterns, relationships, and trends.
6	Create effective and ethical visualizations to communicate data-driven insights clearly.

Contents

Part-A

Unit-1 Introduction to Data Science **8(L) hrs**

Defining Data Science: Basic terminology, The Data Science Venn Diagram, Benefits of Data Science, The Data Science process: problem understanding, data collection, preparation, analysis, modeling and communication, Relation between Data Science and Big Data, Data Science and Business Intelligence.

Unit-2 Statistics for Data Science **8(L) hrs**

Descriptive statistics- mean, median, variance, standard deviation, correlation, Probability basics- random variables, distributions, Sampling, bias, uncertainty, Inferential statistics- estimation, hypothesis testing.

Unit-3 Introduction to R for Data Science **6(L)hrs**

Role of R in data science, features of R, overview of RStudio environment – script, console, workspace and plots; installing and loading packages. Basic syntax and operations: using R as a calculator, assignment operator, comments and basic built-in functions. Data types in R: numeric, integer, character, logical and factor; type checking and type conversion. Core data structures: vectors, matrices, lists and data frames; creating objects and understanding their structure. Working directory and basic file handling commands.

Part-B

Unit-4 Data Collection and Data Representation **8(L)hrs**

Purpose of data collection and defining data requirements, Sources of data: surveys, experiments, transactional systems, sensors, web data, APIs and open data portals, Forms of data: structured, semi structured and unstructured data, Types of data: text, graph, image, sensor and network data, Modern data storage: files, spreadsheets, relational databases, NoSQL databases, distributed databases and data streams, Dataset terminology: observations and variables; discrete and continuous; qualitative and quantitative; dependent and independent variables, Basic data representation using tables, records and metadata, Data quality and ethical issues in data collection.

Unit-5 Data Wrangling and Exploratory Data Analysis **7(L)hrs**

Need for data wrangling in real-world analysis, Data quality issues: missing values, noise, duplicates and inconsistencies, Data cleaning and integration from multiple sources, Data transformation: normalization, scaling, encoding and aggregation, Feature understanding and basic feature extraction, Exploratory Data Analysis (EDA): objectives and importance, summarizing data using descriptive statistics and tables, identifying patterns, trends, relationships and outliers, formulating hypotheses, stating assumptions, and drawing justified conclusions from EDA results.

.

Unit-6 Data Visualization **8(L) hrs**

Purpose and role of visualization in data science, principles of effective visualization: clarity, accuracy and simplicity, common plot types and their usage: bar charts, histograms, box plots, scatter plots and line graphs, visualizing distributions, comparisons and relationships, designing informative charts and dashboards, ethical visualization and avoiding misleading graphics, interpreting and presenting visualizations to support analytical findings, insights, and data-driven decision making.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Note: All experiments shall be carried out as complete data analysis tasks, starting from raw data and ending with interpretation of results. Students must clearly document their approach, assumptions, observations, and conclusions for each experiment.

Experiment No.	Experiment Title
1	To explore the data science process, understand the roles of dependent and independent variables, and illustrate the lifecycle of data analysis using a sample dataset.
2	To compute and interpret basic descriptive statistics including mean, median, variance, and standard deviation using R functions.
3	To analyze relationships between variables using correlation and covariance, and visualize them using scatter plots.
4	To simulate random variables and sampling processes using R to study distributions, sampling bias, and uncertainty, and visualize results using plots.
5	To import data from multiple sources such as CSV/Excel files and SQL databases, and convert them into data frames for analysis using R.
6	To identify data types, perform type conversion, handle missing values, and execute basic data cleaning and transformation operations.
7	To perform data manipulation such as filtering, sorting, grouping, aggregation, joining, and reshaping datasets to prepare tidy data for analysis.
8	To conduct univariate exploratory data analysis including summarization, distribution visualization, and outlier detection.
9	To perform bivariate exploratory data analysis to study relationships between variables, compute correlations, and detect patterns or trends.
10	To apply feature transformations such as normalization, standardization, rescaling, and creation of derived variables to improve data quality and understanding.
11	To create informative visualizations including bar charts, histograms, box plots, scatter plots, and line graphs to explore distributions and relationships.
12	To perform an end-to-end real-world data science case study involving problem definition, data collection, cleaning, exploratory analysis, visualization, and insight generation, with clear explanation of assumptions, methods, and conclusions.

Text Books

1. Sinan Ozdemir and Sunil Kakade, “Principles of Data Science”, Second Edition, Packt Publishing, 2018.
2. Roger D. Peng and Elizabeth Matsui, “The Art of Data Science”, Lean Publishing (Leanpub), First Edition, 2016
3. Davy Cielen, Arno D. B. Meysman, and Mohamed Ali, “Introducing Data Science”, Manning Publications, 1st Edition, 2016.
4. Roger D. Peng, “R Programming for Data Science”, 5th Edition, 2016.

Reference Books

1. Joel Grus, “Data Science from Scratch: First Principles with Python”,, O’Reilly Media, Second Edition, 2019.
2. Cathy O’Neil and Rachel Schutt, “Doing Data Science: Straight Talk from the Frontline”, O’Reilly Media, First Edition, 2014.

3. Trevor Hastie, Robert Tibshirani, and Jerome Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition, Springer. 2017
4. Peter Bruce, Andrew Bruce, and Peter Gedeck, Practical Statistics for Data Scientists, Second Edition, O'Reilly Media. 2020
5. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, First Edition, Wiley.

Online Learning Materials

1. Welcome | R for Data Science — Accessed on 6 Jan, 2026
2. R Programming by Johns Hopkins University | Coursera — Accessed on 6 Jan, 2026
3. A Crash Course in Data Science | Coursera — Accessed on 6 Jan, 2026
4. Data Science | Coursera — Accessed on 6 Jan, 2026

Supplementary Swayam Courses

S. No.	Course Name	Instructor	Host Institute	URL
1.	Exploratory Data Analysis for Data Science with R Software	Prof. Shalabh	IIT Kanpur	Exploratory Data Analysis for Data Science with R Software (English) - Course
2.	Data Science for Engineers	Prof. Ragunathan Rengasamy, Prof. Shankar Narasimhan	IIT Madras	https://onlinecourses.nptel.ac.in/noc21_cs69/preview
3.	Introduction to Data Analytics	Prof. Nandan Sudarsanam, Prof. Balaraman Ravindran	IIT Madras	https://nptel.ac.in/courses/110106072
4.	Scalable Data Science	Prof. Anirban Dasgupta, Prof. Sourangshu Bhattacharya	IIT Gadhinagar, IIT Kharagpur	Scalable Data Science - Course

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS115
Course Title: Digital Forensics

Programme: B.Tech.	**L:**3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L) + 30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – III		

Prerequisites (if any): CCS106, CCS111

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply digital forensic concepts to classify and investigate cybercrimes.
2	Analyze digital crime scenes and implement proper evidence handling procedures.
3	Evaluate forensic artefacts from computers, networks, and mobile devices.
4	Investigate real-world cyber incidents using forensic tools and methodologies.
5	Design legally compliant and ethical digital investigation strategies.

Contents

Part-A

Unit-1 Introduction **8 (L) hrs**

Introduction to Forensic Science, Forensic Science - Traditional vs Digital Forensics, Computer Forensics and Cyber Forensics, Cyber Forensics in Corporate and Law Enforcement Domains, Types of Cybercrimes (Individual, Corporate, Government), The Forensic Investigation Process, Cyber criminalistics and Challenges in Digital Forensics, Importance of digital forensics in the Indian IT industry, Ethics and Professional Responsibilities of Forensic Investigators.

Unit-2 Digital Crime Scene and Evidence Collection **6 (L) hrs**

Digital Crime Scene - Identification and Scope Types of Digital Evidence (Volatile & Non-Volatile), Basic Rules of Evidence and Forensic Principles, Evidence Collection Methods and Tools, Chain of Custody and Evidence Integrity, Seizure of Digital Devices (PCs, Servers, Storage Media), Incident Handling in Corporate Environments.

Unit-3 Computer & File System Forensics **6 (L) hrs**

Disk Forensics Fundamentals, File Systems (FAT, NTFS, EXT), Data Acquisition Techniques (Live & Dead Forensics), Deleted File Recovery and Data Carving, Malware, Spyware, Adware Analysis Basics, Time-stamping, Logs, and Timeline Analysis.

Part-B

Unit -4 Network & Mobile Forensics **6 (L) hrs**

Basics of Network Forensics, Network Forensics Concepts and Architecture Logs, Traffic Analysis, IDS/IPS Logs, Email and Web Forensics, Wireless Forensics, Mobile Forensics: Android & iOS Basics SIM, Call Logs, Messages etc.

Unit -5 Impairment (Anti-Forensic) Tools **7 (L) hrs**

Introduction to Impairment (Anti-Forensics), types and objectives of impairment tools - DBAN, SDelete, BleachBit, Steghide, VeraCrypt, Event log cleaners, CCleaner, ExifTool.

Unit -6 Digital Forensics Tools **7 (L) hrs**

Open-Source Tools - Honeypots, Autopsy, Python, Volatility, Wireshark, NetworkMiner, Plaso/Log2Timeline, Bulk Extractor and Commercial Tools - EnCase (by OpenText), FTK (Forensic Toolkit), X-Ways Forensics, Cellebrite, Oxygen.

Unit-7 Emerging Trends **5 (L) hrs**

Emerging Trends - Cloud Forensics, IoT Forensics, AI in Digital Forensics, Blockchain Forensics, Big Data Forensics.

Laboratory Work

In the laboratory work, students will implement programs related to the above topics using any open-source.

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	**Cybercrime Case Analysis:** Analyze real cybercrime cases and identify the type of cybercrime and investigation steps.
2.	**Forensic Investigation Process Mapping:** Prepare a flowchart of the digital forensic investigation process with ethical guidelines.
3.	**Digital Evidence Identification:** Identify volatile and non-volatile data from a running computer system.

4.	**Chain of Custody Preparation:** Create a sample chain-of-custody document for seized digital evidence.
5.	**Evidence Collection Demonstration:** Demonstrate safe shutdown and evidence collection procedure for a computer system.
6.	**Disk Imaging Practical:** Create a forensic image of a storage device using a forensic imaging tool.
7.	**Deleted File Recovery:** Recover deleted files from a storage device and document the findings.
8.	**Log and Timeline Analysis:** Analyze system log files and create a basic activity timeline.
9.	**Network Traffic Analysis:** Capture and analyze basic network traffic using a network analysis tool.
10.	**Email and Browser Forensics:** Analyze email headers and browser history for forensic information.
11.	**Mobile Forensics Demonstration:** Perform a basic demonstration of mobile data extraction (calls, SMS, app data).
12.	**Study of Anti Forensics Tools:** Study the concept, types, and objectives of impairment (anti-forensic) tools.
13.	**Study of Digital Forensics Tools:** Identify and study basic digital forensic tools used in computer engineering.
14.	Understand attack monitoring using Honeypot cyber systems.

Text Books

1. B. Nelson, A. Phillips, C. Steuart, “Guide to Computer Forensics and Investigations”, Cengage, 7th Edition, 2003.
2. John R. Vacca, “Computer Forensics: Computer Crime Scene Investigation”, Laxmi Publications, 1st Edition, 2015.
3. Gerard Johansen, “Digital Forensics and Incident Response-Digital Forensics and Incident Response”,, ISBN: 978-1838649005, 2nd Edition, 2020.
4. Chris Prosise, Kevin Mandia, “Incident Response & Computer Forensics”, McGraw-Hill, 2nd Edition, 2003.
5. Cory Altheide & Harlan Carvey, “Digital Forensics with Open Source Tools”, Publisher: Syngress (Elsevier), 1st Edition, 2011.
6. Suryadipta Majumdar, Paria Shirani, Lingyu Wang, “Innovations in Digital Forensics”, World Scientific Publishing, ISBN: 978-9811273216, 1st edition, 2023.

Reference Books

1. John Sammons, “The Basics of Digital Forensics”, Elsevier, 2nd Edition, 2012.
2. Ravi Kumar & B. Jain, “Cyber Forensics: Concepts and Approaches”, 2006.

3. Brian Carrier, "File System Forensic Analysis", Addison-Wesley Professional, 1st Edition, 2005.
4. Michael Hale Ligh, Andrew Case, Jamie Levy, AAron Walters, "The Art of Memory Forensic", Wiley, 1st Edition, 2014.
5. Anita Gehlot, Rajesh Singh, Jaskaran Singh, Neeta Raj Sharma, "Digital Forensics and Internet of Things", John Wiley & Sons, ISBN: 978-1-119-76878-4, 1st Edition, 2022.

Online Learning Materials

1. https://onlinecourses.swayam2.ac.in/e-learning/preview/nou24_cs04 Accessed on 5-1-2026
2. https://onlinecourses.nptel.ac.in/e-learning/preview/noc24_cs121 Accessed on 5-1-2026
3. https://www.wireshark.org/ Accessed on 5-1-2026

Supplementary SWAYAM Courses

S. No.	Course Name	Instructor	Host Institute	URL
1.	Digital Forensic	By Dr. Jeetendra Pande	Uttarakhand Open University, Haldwani	https://onlinecourses.swayam2.ac.in/nou25_cs05/preview
2.	Cyber Security and Privacy	By Prof. Saji K Mathew	IIT Madras	https://onlinecourses.nptel.ac.in/noc23_cs127/preview
3.	Forensic Linguistics	By Prof. Deepak Mashru	National Forensic Sciences University (INI)	https://onlinecourses.nptel.ac.in/noc25_hs89/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS116
Course Title: IoT Architecture and Protocols

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45 (L)+30 (P)= 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 30%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – III		

Prerequisites (if any): CCS106

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand the fundamental concepts, components and evolution of IoT architectures.
2	Analyze different IoT architectural models and their suitability for applications.
3	Describe and compare IoT communication protocols across different layers.
4	Apply and analyze IoT communication models, APIs, and technologies to select suitable communication solutions.
5	Interpret the concept of interoperability, scalability and security considerations in IoT systems.
6	Design and implement an IoT application using standard platforms, tools, and design methodology.

Contents

Part-A

Unit-1 Introduction to Internet of Things **9(L) hrs**

Internet of Things (IoT) - Definition, Vision. Characteristics IoT ecosystem - devices/sensors, gateways, cloud. Users Domain specific applications of IoT - Home automation, Industry applications, Surveillance applications, Environmental and Agriculture applications. Challenges in IoT - scalability, interoperability, energy efficiency, Smart and Hyper-Connected Devices, Conceptual Framework, Major Components of IoT System, Overview of IoT reference model.

Unit-2 IoT Architecture Models **7(L) hrs**

Layered IoT architecture (3-layer, 5-layer model), Device layer, Network layer, Application layer Cloud-centric and Edge-centric architectures, Comparison of different architectural approaches. Fog

and Edge computing in IoT - Data processing, advantages, challenges and applications. Developing IoT solutions using different IoT tools.

Unit-3 IoT Communication Technologies **7(L) hrs**

IoT Communication Models: Request–Response, Publish–Subscribe, Push-Pull and Exclusive Pair. IoT communication APIs - REST-based and WebSocket-based. Short-range communication - RFID, NFC, Bluetooth, ZigBee. Long-range communication - LoRaWAN, NB-IoT, Sigfox, Wired vs wireless communication in IoT, Role of gateways and brokers, Selection criteria for communication technologies.

Part-B

Unit-4 IoT Protocols **9(L) hrs**

Link Layer - Ethernet, WiFi, WiMax, LR-WPAN. Network Layer - IPv4, IPv6, 6LoWPAN Transport layer protocols - TCP. UDP Application layer protocols - HTTP, HTTP/REST, CoAP, MQTT, SMQTT, AMPQ, XMPP, DDS. WebSocket Routing protocols - RPL Device management and discovery, Interoperability and standardization, Overview of IoT platforms and middleware.

Unit-5 Security, Interoperability & Case Studies **6(L) hrs**

Security challenges in IoT, Authentication, authorization, and data integrity. Secure communication protocols - Introduction to secure communication in IoT networks. Overview of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS), their working principles, differences, and suitability for constrained IoT devices. Interoperability issues and solutions. Privacy and Security Issues in IoT - data collection, storage, and sharing in IoT applications. Case-based discussion of privacy and security issues in real-world IoT applications.

Unit-6 IoT Platform Design Methodology **7(L) hrs**

IoT Design Methodology, Constraints of IoT Systems - Constraints affecting design of IoT, Technical design Constraints. IoT Design Life Cycle - Purpose and requirement specification, Process specification, Domain Model specification, Information Model specification, Service specification, IoT level specification, Functional view specification, Operational view specification, Device and component Integration and Application Development. Case Study: Design of a Simple IoT Application (Smart Home / Weather Monitoring).

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1.	To study and understand the basic components of an IoT system and its layered architecture using any simulation tool to identify IoT components: sensors, actuators and gateways.

2.	Familiarization with Arduino/ Raspberry Pi and perform necessary software installation.
3.	Interface LED/ Buzzer with Arduino/ Raspberry Pi and write a program to turn ON/OFF LED for specific duration.
4.	Interface DHT11/ DHT22 sensor with Arduino/ Raspberry Pi and write a program to print temperature and humidity readings.
5.	To simulate communication between IoT devices for creating a basic IoT structure using a software tool.
6.	To study and compare short-range and long-range IoT communication technologies such as Bluetooth, ZigBee, Wi-Fi, LoRaWAN, and NB-IoT based on communication range, data rate, power consumption, and application use cases.
7.	To implement the CoAP/MQTT publish–subscribe communication model using a suitable software tool.
8.	To implement any IoT communication protocols using suitable APIs.
9.	To log data using Raspberry PI and upload to any cloud platform.
10.	To prepare a mini project based on an IoT system to study and design an IoT architecture for a real-world application: • Choose an application (Smart Home, Smart Agriculture, Healthcare, etc.), • Design complete architecture, • Select and apply the appropriate communication protocols.

Text Books

1. Davis Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton, and Jerome Henry, "IoT Fundamentals – Networking Technologies, Protocols, and Use Cases for the Internet of Things", CISCO Press, 5th Impression, 2014.
2. Mayur Ramgir, "Internet of Things – Architecture, Implementation and Security', 1st Impression, Pearson India, 2019.
3. Raj Kamal, "Internet of Things – Architecture and Design Principles", McGraw - Hill Education, 5th Reprint Edition, 2017.
4. Arsheep Bahga, Vijay Madisetti, "Internet of Things – A Hands-On Approach", Universities Press, 3rd Impression,, 2015.
5. Gaston C. Hillar, "Internet of Things with Python", PACKT Open Source Press, 2nd Impression, 2016.

Reference Books:

1. Rajesh Singh, Anita Gehlot, Lovi Raj Gupta, Bhupendra Singh, Mahendra Swain, "Internet of Things with Raspberry Pi and Arduino", CRC Press, 1st Impression, 2015.
2. Ashwin Pajankar, "Internet of Things with Arduino and Bolt", BPB Publications, 2nd Impression, 2018.

Online Learning Materials

1. Donald Noris -The Internet of Things, Mc Graw Hill Education, New York https://www.pdfdrive.com/download.pdf?id=176037121&h=c064c3b7014c0ed46f60e480e4a5b625 & u=cache&ext=pdf Accessed on 5-1-2026
2. Aleksandr Kapitonov, Raivo Sell - Introduction to Internet of Things, Erasmus Education, Ritankar Sahu Publisher http://iot-open.eu/download/io1-introduction-to-theiot/?wpdmdl=2702&_wpdmkey=6022469db61f3&refresh=6022469dbfe3a1612859037 Accessed on 5-1-2026

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Introduction to Internet of Things	By Prof. Sudip Misra	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc26_cs37/preview
2	Internet of Things: Design Concepts and Use Cases	By Prof. Maitreyee Dutta	NITTTR, Chd.	https://onlinecourses.swayam2.ac.in/ntr26_ed23/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS117
Course Title: Software Testing and Quality Assurance

Programme: B.Tech.	**L:** 3 **T:** 0 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical:** Integrated	**Teaching Hours:** 45(L)+30(P) =75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 10%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – III		

Prerequisites (if any): CCS112

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Analyze fundamental software engineering and software testing concepts.
2	Design effective test cases using black-box and white-box testing techniques.
3	Implement unit testing using automated testing tools.
4	Analyze integration, system, acceptance, performance and security testing using appropriate methods.
5	Examine software quality and quality assurance practices using standard quality models, metrics, and process maturity frameworks.

Contents

Part-A

Unit -1 Introduction **7(L) hrs**

Role of Testing in Modern Software Development, Verification and Validation, Test Case, Expected Outcome, Issues in Testing, Testing Activities, Test Levels, Sources of Information for Test Case Selection, Test Planning and Design, Monitoring and Measuring, Test Execution, Test Tools and Automation, Test Team Organization and Management. Quality Engineering (QE), Quality Assurance (QA), Quality Control (QC), Quality Metrics.

Unit-2 Testing Techniques and Types **12(L) hrs**

Unit Testing - Static Unit Testing, Dynamic Unit Testing, Defect Prevention, Mutation Testing, Debugging, Black Box Testing- Equivalence Class Partitioning, Boundary Value Analysis, Orthogonal

Array Testing, White-Box Testing-Statement Coverage, Branch Coverage, Multiple Condition Coverage, Path Coverage, Control Flow Testing- Control Flow Graph, Paths in a Control Flow Graph, Path Selection Criteria, Data Flow Testing- Data Flow Anomaly, Data Flow Graph, Data Flow Testing Criteria. Domain Testing- Domain Error, Testing for Domain Errors, Types of Domain Errors, ON and OFF Points, Unit Testing in eXtreme Programming, JUnit: Framework for Unit Testing, Tools for Unit Testing, Integration Testing- Different Types of Interfaces and Interface Errors, System Integration Techniques (Incremental, Top Down, Bottom up, Sandwich and Big Bang), Software and Hardware Integration, Hardware Design Verification Tests, Hardware and Software Compatibility Matrix**.**

Unit-3 Test Design and Execution **8(L) hrs**

The Seven-Step Software Testing Process, Test Design Factors, Characteristics of Testable Requirements, Modeling a Test Design Process, Modeling Test Results, Test Design Preparedness Metrics, Structure of a System Test Plan, Test Suite Structure, Test Effort Estimation, System Test Automation, Metrics for Tracking System Test, Types of Acceptance Testing, Acceptance Criteria, Performance Testing basics: Load, stress, scalability, Security Testing essentials: OWASP (Open Web Application Security Project), Vulnerability scanning.

Part-B

Unit-4 Software Quality **10(L) hrs**

Five Views of Software Quality, McCall's Quality Factors and Criteria**,** Quality Factors**,** Quality Criteria, Relationship between Quality Factors and Criteria, Quality Metrics, ISO 9126 Quality Characteristics, ISO 9000:2000 Software Quality Standard, ISO 9000:2000 Fundamentals, ISO 9001:2000 Requirements, Capability Maturity Model(CMM), CMM Architecture, Five Levels of Maturity and Key Process Areas, Common Features of Key Practices, Application of CMM, SEI Capability Maturity Model, Capability Maturity Model Integration (CMMI), Test Process Improvement, Testing Maturity Model. Software Process Improvement and Capability Determination (SPICE), Personal Software Process (PSP).

Unit-5 Quality Assurance **8(L) hrs**

Software Quality Assurance (SQA) Versus Traditional QA, Software Safety and Its Relation to Software Quality Assurance, Development Quality Assurance, Quality Management in IT, Introduction to ITIL (Information Technology Infrastructure Library), Software Quality Assurance Metrics, Software Benchmarks and Baselines. DevOps Quality Metrics – DORA (DevOps Research and Assessment), Accelerate Metrics, Continuous Compliance and Governance, AI in SQA for Risk Prediction & Test Optimization**.** Total Quality Management, Statistical Process Control (SPC), Six Sigma, Plan-Do-Check-Act (PDCA).

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Experiment No.	Experiment Title
1	Study various Open Source Software Testing Tools (Junit, Selenium)
2	Demonstrate working in JUnit: Framework for Unit Testing.
3	Analyze source code to demonstrate equivalence class partitioning and boundary value analysis techniques.
4	Analyze source code to demonstrate path coverage, statement coverage and branch coverage techniques.
5	Design and implement a modular test automation framework using Selenium with POM (Page Object Model) and data-driven testing.
6	Demonstrate mutation testing in evaluating the strength of existing unit tests.
7	Demonstrate working of OWASP ZAP for Security Testing and vulnerability assessment.
8	Demonstrate integration of automated testing in CI/CD pipeline.
9	Study McCall's quality factors and DORA metrics.
10	Case study: Students prepare case study reports to showcase how companies can improve products/services by implementing Quality Assurance strategies, often involving automation, process overhauls and using new methodologies.

Text Books

1. Kshirasagar Naik, Priyadarshi Tripathy, "Software Testing and Quality Assurance-Theory and Practice", John Wiley & Sons, Inc., 2008, ISBN 978-0-471-78911-6", Second edition, 2008.
2. G. Gordon Schulmeyer, "Handbook of Software Quality Assurance", ARTECH HOUSE, INC., ISBN-13: 978-1-59693-186-2, Fourth Edition, 2008.
3. William E. Perry, "Effective Methods for Software Testing", John Wiley and Sons, ISBN 9971-51-345-5, Third edition, 2005.

Reference Books

1. Fenton, Pfleeger, "Software Metrics: A Rigorous and Practical Approach", Thomson Brooks/Cole, ISBN 981-240-385-X, Third edition, 1997.
2. Desikan, Ramesh, "Software Testing: Principles and Practices", Pearson Education, ISBN 81-7758-121-X, 2006, sixth edition, 2007.
3. Anne Mette, Jonassen Hass, "Guide to Advanced Software Testing", ARTECH HOUSE, INC., 2008, ISBN-13: 978-1-59693-285-2, Second Edition, 2008.

Online Learning Materials

1. Software Testing and Quality Assurance, Theory and Practice by Kshirasagar Naik and Priyadarshi Tripathy.https://www.softwaretestinggenius.com/download/staqtpsn.pdf Accessed on Dec 09, 2025
2. https://www.softwaretestinggenius.com/download/EMFST.pdf Accessed on Dec 11, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Software Testing (Black-box & White-box Methods)	Prof. Rajib Mall (IIT Kharagpur)	SWAYAM-hosted NPTEL course	https://onlinecourses.nptel.ac.in/noc25_cs66/preview
2	Software Testing	Prof. Meenakshi D'souza (IIIT Bangalore)	SWAYAM-hosted NPTEL course	https://onlinecourses.nptel.ac.in/noc22_cs61/preview
3	Quality Design and Control (Quality Assurance Concepts)	Prof. Pradip Kumar Ray, IIT Kharagpur	SWAYAM-hosted NPTEL course	https://onlinecourses.nptel.ac.in/noc20_mg18/preview

Guru Nanak Dev Engineering College, Ludhiana

An Autonomous College under UGC Act 1956

B.Tech. (CSE), Scheme - 2024

Course Code: ECS118
Course Title: Computer Vision

Programme: B.Tech.	**L:** 3 **P:** 2	**Credits:** 4
Semester: 6	**Theory/Practical/ Integrated:** Integrated	**Teaching Hours:** 45(L)+30(P) = 75 hrs
Total Max. Marks: 150	**Continuous Assessment (CA) Marks:** 90	**End Semester Examination (ESE) Marks:** 60
Minimum Percentage of Numerical / Design / Programming Problems in ESE: 50%		
Duration of End Semester Examination (ESE): 3 hours		
Course Type: Elective – III		

Prerequisites (if any): BSCS101, ECS106, CCS116

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Demonstrate understanding of fundamental concepts of computer vision.
2	Apply multi-view geometry and depth estimation techniques for 3-D scene analysis.
3	Extract visual features and descriptors for effective representation of images and scenes.
4	Implement object detection and pattern recognition techniques using clustering and classification methods for real-world vision problems.
5	Design motion analysis algorithms for dynamic scene interpretation.

Contents

Part-A

Unit-1 Introduction to Computer Vision **6(L)hrs**

Overview of Computer Vision, Applications of Computer Vision in Robotics, Healthcare, and Autonomous Vehicles, Basic Image Formation- Geometric primitives and transformations (2D and 3D transformations), Photometric Image Formation – Lighting, Reflectance and Shading, Optics.

Unit-2 Depth estimation and Multi-camera views **8(L)hrs**

Depth perception, Binocular Stereopsis, Epipolar Geometry, 3-D reconstruction framework, Rectification, Direct Linear Transformation (DLT), Random Sample Consensus (RANSAC) algorithm, Homography.

Unit-3 Visual Feature Extraction and Representation **9(L) hrs**

Introduction to type of features, Boundary Feature Descriptors, Region Feature Descriptors, Principal Components as Feature Descriptors, Whole Edges- Canny, Laplacian of Gaussian (LoG), Difference of Gaussian (DoG), Hough Transform, Corners- Harris-Stephens corner detector, Whole Image Features- SIFT (Scale Invariant Feature Transform), LBP (Local Binary Pattern) and SURF.

Part-B

Unit-4 Object Detection & Pattern Analysis **12 (L) hrs**

Object Recognition - Patterns and Patterns classes, Pattern Classification by Prototype Matching, Basics of Probability and Statistics, Clustering: K-Means, K-Medoids, Mixture of Gaussians, Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised. Classifiers - Optimum (Bayes) classifier, k-Nearest Neighbor, Neural Network models.

Unit-5 Motion Analysis **10 (L) hrs**

Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo, Motion parameter estimation.

Laboratory Work

The following is only the suggested list of Practical's. Instructors may frame additional Practical's relevant to the course contents.

Students can implement various techniques and algorithms using MATLAB/Python/Octave:

Experiment No.	Experiment Title
1	Analyze the image formation process and observe the effects of 2D and 3D geometric transformations on digital images using basic geometric primitives.
2	Illustrate the effects of lighting, reflectance, shading, and optical characteristics on the photometric image formation process.
3	Compute depth information from stereo image pairs using disparity estimation.
4	Estimate Epipolar geometry between two views.
5	Perform homography-based 3D reconstruction using DLT and RANSAC and evaluate reconstruction accuracy under noisy conditions.
6	Implement Canny, Laplacian of Gaussian (LoG), and Difference of Gaussian (DoG) edge detection techniques on digital images.
7	Detect corner features in digital images using the Harris–Stephens corner detector.
8	Extract image features using SIFT and LBP descriptors and reduce feature dimensionality using Principal Component Analysis (PCA).

9	Represent objects as feature vectors and analyze pattern classes.
10	Classify patterns using prototype matching and analyze statistical properties of pattern classes.
11	Implement K-Means, Mean-Shift, and Gaussian Mixture Models (GMM) for pattern analysis.
12	Implement Bayes, KNN, and ANN classification algorithms on reduced-dimensional image feature sets.
13	Implement background subtraction and optical flow algorithms to detect moving objects in video data.
14	Implement KLT tracker and perform spatio-temporal analysis to estimate motion parameters in dynamic scenes.

Text Books

1. Szeliski, R., "Computer Vision: Algorithms and Applications", Springer-Verlag London Limited, 1st Edition, 2011.
2. Hartley, R. and Zisserman, A., 'Multiple View Geometry in Computer Vision Cambridge", University Press, 2nd Edition, 2003.
3. Gonzalez, C., R. and Woods, E., R.," Digital Image Processing", Addison- Wesley, 4th Edition, 2018.

Reference Books

1. Forsyth, A., D. and Ponce, J., "Computer Vision: A Modern Approach", Pearson Education, 2nd Edition, 2012.
2. Fukunaga, K., "Introduction to Statistical Pattern Recognition", Academic Press, Morgan Kaufmann, 2nd Edition, 1990
3. Low A., "Introductory Computer Vision and Image Processing", McGraw-Hill, 1st edition, 1991.

Online Learning Materials

1. https://www.coursera.org/learn/introduction-computer-vision-watson-opencv
   Accessed on Dec 27, 2025
2. https://youtu.be/3LaVxEX3F0o Accessed on Dec 27, 2025

Supplementary SWAYAM Course

Sr. No.	Course Name	Instructor	Host Institute	URL
1	Computer Vision	By Prof. Jayanta Mukhopadhyay	IIT Kharagpur	https://onlinecourses.nptel.ac.in/noc19_cs58/preview

2	Computer Vision and Image Processing - Fundamentals and Applications	Prof.M.K. Bhuyan	IIT Guwahati	https://onlinecourses.nptel.ac.in/noc26_ee31/preview