Sl. No.	**Content**
1	**Computer Science Courses for Undergraduate Programme of study with Computer Science discipline as one of the three Core Disciplines** DISCIPLINE SPECIFIC ELECTIVES (DSE) **Sem 4:** 1) DSE 02a: Data exploration and visualization 2) DSE 02b: Software Engineering **Sem 5:** 1) DSE 03a : Data Mining for Knowledge Discovery 2) DSE 03b: Web Design and Development **Sem 6:** 1) DSE 04a: Applied Network Analytics DSE 04b: Foundations of Computer Graphics

Department of Computer Science

COURSES OFFERED BY DEPARTMENT OF COMPUTER SCIENCE

(Computer Science Courses for Undergraduate Programme of study with Computer Science discipline as one of the three Core Disciplines)

Semester 4

DISCIPLINE SPECIFIC Elective - (DSE-2a) : Data Exploration and Visualization

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 02a Data Exploration and Visualization	**4**	**3**	**0**	**1**	Pass in Class XII	Basic Knowledge of Python

Learning Objectives

The course is designed to equip students with the skills to analyze diverse real-world data to extract data insights and interpret results. Subsequently, they will learn to apply exploratory data science techniques to effectively communicate findings, using Python for effective problem-solving.

Learning Outcomes

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

- Create and manipulate NumPy arrays to perform data analysis.
- Use Pandas methods to import, export, and preprocess data from various sources.
- Perform basic data manipulation tasks, including data cleaning, filtering, sorting, and merging on Pandas objects .

- Use grouping and aggregation operations in Pandas to summarize data in Series and DataFrame objects; and analyze and interpret data based on grouped and aggregated results.
- Use Matplotlib and Seaborn to create static visualizations and plotly to create interactive visualizations of data to communicate data insights.

SYLLABUS OF DSE 02 a

Unit 1 (10 Hour): Creating and Manipulating NumPy arrays: creating arrays, indexing and slicing, mathematical operations with NumPy arrays

Unit 2(15 Hours): Data Manipulation with Pandas: Series and DataFrame objects; importing and exporting data from various file formats into pandas DataFrame; Data selection and filtering-indexing, slicing, conditional filtering using boolean indexing; Data Cleaning- handling missing data in Pandas and outlier detection; Data Manipulation-sorting, reshaping, merging.

Unit 3 (5 Hours): Grouping and Aggregation with Pandas: Grouping data using Pandas, applying aggregation functions such as sum, mean, count, etc.to grouped data, using pivot tables and cross-tabulation for data summarization

Unit 4 (10 Hours): Data Visualization with Matplotlib and Seaborn: Introduction to Matplotlib and Seaborn to plot data using figures and subplots, Plots - Line plots, scatter plots, and bar plots, Visualizing distributions using histogram and box plots, Customizing plot aesthetics and adding annotations

Unit 5 (5 Hours): Interactive Visualizations with Plotly: Introduction to Plotly library for interactive visualization; Creating interactive line plots, scatter plots, and bar plots; Adding interactivity with hover effects, zooming, and panning

Essential/recommended readings

1. VanderPlas, J. Python data science handbook: Essential tools for working with data. " O'Reilly Media, Inc.", 2nd edition.

2. McKinney W. Python for Data Analysis: Data Wrangling with Pandas, NumPy and IPython. 2nd edition. O'Reilly Media, 2018.

3. Molin S. Hands-On Data Analysis with Pandas, Packt Publishing, 2019.

4. Rahman K. Python Data Visualization Essentials Guide: Become a Data Visualization expert by building strong proficiency in Pandas, Matplotlib, Seaborn, Plotly, Numpy, and Bokeh, BPB 2021

Additional References:

1. Chen D. Y, Pandas for Everyone: Python Data Analysis, Pearson, 2018.

Online references/material:

1. https://www.indeepdata.com/blog/exploratory-data-analysis/

Suggested Practical List (If any): (30 Hours)

Use data set of your choice from Open Data Portal (https:// data.gov.in/, UCI repository) or load from scikit, seaborn library for the following exercises to practice the concepts learnt.

1. Write a program using the NumPy library to perform the following tasks:

    A. Generate a 5x2 integer array with values ranging from 50 to 100, where each element has a difference of 5. Reshape the resulting array to a size of 10x1.
    B. Create a 1D random array with values ranging from 1 to 100. Calculate various statistical measures such as minimum, maximum, mean, median, standard deviation, number of unique values, count of unique values, and the most frequent value in the array.
    C. Create a 5x5 identity matrix where all the diagonal elements are set to the value 5.
    D. Consider a dataset containing the heights (in centimeters) and weights (in kilograms) of 20 individuals. Your task is to perform various operations using the NumPy library to analyze the data.
        a. Create a NumPy array called "heights" with the following height values: [165, 170, 175, 168, 172, 180, 160, 169, 176, 171, 174, 182, 158, 167, 173, 179, 163, 166, 177, 181]. Create a NumPy array called "weights" with the following weight values: [60, 65, 70, 75, 80, 85, 55, 58, 63, 68, 72, 77, 50, 62, 67, 74, 52, 57, 69, 73].
        b. Create a new NumPy array called "combined" by stacking the heights and weights arrays such that the shape of the resulting array is 20 x 2.
        c. Calculate and print the mean height and weight of the individuals in the dataset.
        d. Find and print the index of the shortest and tallest individuals in the dataset.
        e. Sort the array based on height on the individuals.
        f. Swap the positions of the two columns in the array.
        g. Retrieve records of individuals having weight below 70kg.

2. Write a program using the Pandas library to perform the following operations on the penguins dataset from the Seaborn library:

A. Load the penguins dataset into a Pandas dataframe.
B. Determine the number of observations/records and the number of attributes in the dataframe.
C. Display the names of the attributes, row indexes, and data types of each attribute in the dataframe.
D. Display the first 5 and last 5 records of the dataframe.
E. Retrieve the values of the second column for the third and fourth records.
F. Display a summary of the data distribution for all attributes in the dataframe.
G. Compute the pairwise correlation between all attributes in the dataframe.

3. Consider the Titanic dataset, which contains information about passengers on board the Titanic, including their age, gender, passenger class, survival status, and other attributes. Write a program using the Pandas library to perform the following operations on the Titanic dataset:

A. Load the Titanic dataset into a Pandas DataFrame.
B. Check for any duplicate records and missing values in the dataset and handle them appropriately.
C. Calculate and display the total number of passengers who survived and those who did not.
D. Filter the DataFrame to select only the records of passengers who were under the age of 18.
E. Calculate the average age for passengers belonging to each of the passenger class.
F. Create a new column in the DataFrame called "Family Size" that represents the total number of family members (including the passenger) on board.
G. Calculate the correlation between age and fare attributes of the dataset.
H. Create a contingency table that shows the count of passengers based on their survival status (survived or not) and passenger class (first, second, or third class). for titanic dataset

4. Utilize the iris dataset from the Sklearn library to generate various visual representations of the data using the Matplotlib and or Seaborn libraries with proper legends and labels. Perform the following tasks:

A. Create a scatter plot to visualize the relationship between petal length and petal width for different instances of iris flowers.
B. Generate histograms to display the data distribution of each of the four attributes in the iris dataset.
C. Construct a pie chart to illustrate the frequency count of each flower type in the iris dataset.

D. Create a pair plot that showcases the relationship between every pair of attributes in the iris dataset (only seaborn library).

5. Create the visualizations of question 4 using plotly library.

DISCIPLINE SPECIFIC Elective - (DSE-2b) : Software Engineering

Semester 4

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 02b Software Engineering	**4**	**3**	**0**	**1**	Pass in Class XII	**Knowledge of any programming Language**

Learning Objectives

The course introduces software development life cycle and software project management. It includes requirement engineering, software project planning, software design, software quality and software testing.

Learning Outcomes

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

- Understand and apply the software process models.
- Elicitate and analyse customer requirements and map requirements to design models.
- Estimate size, effort and cost required to build software.
- Identify risks involved in the software development and understand software quality.
- Design test cases to perform software testing.

SYLLABUS OF DSE 02b

Unit 1 (9 Hours)

Introduction Software Process, Software Characteristics, Changing nature of Software, Role of Management in Software Development; Software Life Cycle Models- Waterfall Model, Incremental Process Model, Prototyping Model, and Spiral Model; Introduction to Agile development.

Unit 2 (6 Hours)

Requirement Engineering: Crucial Process steps, Types of Requirements, Requirement Elicitation, Requirement Analysis, Requirement Documentation, Requirement Validation.

Unit 3 (7 Hours)

Software Project Planning: Size Estimation – Lines of code, Function points; Cost estimation using Constructive Cost Model (COCOMO model), Software Risk Management, Project Scheduling (Gantt Chart).

Unit 4 (8 Hours)

Software Design: Conceptual and Technical Design, Objectives of Design, Module Coupling and Cohesion, Strategy of Design, Function-oriented Design

Unit 5 (5 Hours)

Software Quality: Quality Attributes, McCall Software Quality Model, Capability Maturity Model (CMM), Software Reliability.

Unit 6 (10 Hours)

Software Testing: Error, Fault and Failure, Functional Testing- Boundary Value Analysis, Equivalence Class Testing; Structural Testing- Path Testing, Cyclomatic Complexity; Levels of Testing, Validation Testing.

Essential/recommended readings

1. RS Pressman, BR Maxim, Software Engineering: A Practitioner's Approach, 8th Edition, McGraw-Hill, 2015.

2. K.K Aggarwal, Yogesh Singh, Software Engineering, 3rd Edition, New Age International Publishers, 2007

Additional References

1. Ken Schwaber, Jeff Sutherland, The Definitive Guide to Scrum: The Rules of the Game, July 2016. [https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf]
2. I Sommerville, Software Engineering, 9th edition, Addison Wesley, 2011.
3. P Jalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House, 2005.

Suggestive Practice Questions:

Create a project report that includes the following –

1. Define Problem Statement for the project and identify process model.
2. Requirement Elicitation, Requirement Analysis: Data Modelling, Software Requirement Specification Document.
3. Project Management: Compute Function Point, Effort, Cost, Risk Identification, Gantt Chart.
4. Software Design: Structured Chart, Pseudocode of a small module.
5. Coding: Develop at least a single module using any programming Language.
6. Testing: Compute Cyclomatic Complexity for the coded module, and generate test cases.

Some of the Sample Projects are given below though they are not limited to this.

1. College Canteen Automation System
2. Online Car-Pooling System
3. Patient Appointment System
4. Medical Prescription Processing System
5. Online Shopping Management System
6. Online Hotel Reservation Service System
7. Online Movie Booking System
8. Online Examination and Result computation System
9. Automatic Internal Assessment System
10. Complaint Ticket Management System

Department of Computer Science

COURSES OFFERED BY DEPARTMENT OF COMPUTER SCIENCE

(Computer Science Courses for Undergraduate Programme of study with Computer Science discipline as one of the three Core Disciplines)

Semester 5

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 03a : Data Mining for Knowledge Discovery	**4**	**3**	**0**	**1**	Pass in Class XII	**Knowledge of Python Programming**

Learning Objectives

This course aims to introduce supervised and unsupervised data mining techniques, and their applications to real-life datasets. The students will learn about data quality, how to pre-process a dataset to make it ready for the application of data mining algorithms. The course will primarily focus on two core data mining techniques: classification and clustering. Different algorithms under the ambit of these techniques will be discussed, along with their strengths and weaknesses, model evaluation, and result evaluation metrics. The importance of ensemble methods, random forests, and the use of bagging and boosting in ensembles will be explained. The students will be

encouraged to apply the aforementioned data mining concepts to real life problems using open-source software.

Learning outcomes

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

1. Pre-process the data for subsequent data mining tasks
2. Apply a suitable classification algorithm to train the classifier and evaluate its performance.
3. Apply appropriate clustering algorithm to cluster the data and evaluate clustering quality
4. Differentiate between partition-based, density-based and hierarchical clustering
5. Build ensemble models to improve predictive performance of a classifier
6. Apply appropriate data mining techniques to solve a real-life problem

SYLLABUS OF DSE 03a

Unit 1 Introduction (6 Hours)

Need for data mining, Data mining tasks, Applications of data mining, Measures of similarity and dissimilarity, Supervised vs. unsupervised techniques.

Unit 2 Data collection and preparation (8 Hours)

Measurement and data collection issues, Data aggregation, Sampling, Dimensionality reduction, Feature subset selection, Feature creation, Discretization and binarization, Variable transformation..

Unit 3 Clustering data (14 Hours)

Basic concepts of clustering, Partitioning Methods: K-means algorithm, Hierarchical Methods: Agglomerative Hierarchical Clustering, Density-Based Methods: DBSCAN Algorithm, Strengths and weaknesses of different methods, Cluster evaluation.

Unit 4 Classification (10 Hours)

Preliminaries, Naive Bayes classifier, Nearest Neighbour classifier, Decision tree, Artificial Neural Network, overfitting, Confusion matrix, Evaluation metrics and Model evaluation.

Unit 5 Ensemble Methods (7 Hours)

Need for ensembles**,** Random Forest, Concept of Bagging and Boosting in ensembles.

Essential/recommended readings

1. Tan P.N., Steinbach M, Karpatne A. and Kumar V. *Introduction to Data Mining*, Pearson, 2019.
2. Zaki M. J. and Meira J. Jr. *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*, 2nd edition, Cambridge University Press, 2020.
3. Aggarwal C. C. *Data Mining: The Textbook*, Springer, 2015.

Additional References:

1. Han J., Kamber M. and Pei J. *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers. 2011.
2. Dunham M. *Data Mining: Introductory and Advanced Topics*, Pearson, 2006.

Online references/material:

1. http://www.dcc.fc.up.pt/~ltorgo/DM1/dataPreProc.html
2. https://www.coursera.org/specializations/data-mining-foundations-practice

Suggested Practical List (If any): (30 Hours)

All topics covered in theory will be implemented using Python. The operations may be performed on the datasets loaded through scikit, seaborn libraries or can be downloaded from from Open Data Portal (https:// data.gov.in/, UCI repository http://archive.ics.uci.edu/ml/ **).**

Recommended Datasets for :

Classification: Abalone, Artificial Characters, Breast Cancer Wisconsin (Diagnostic)

Clustering: Grammatical Facial Expressions, HTRU2, Perfume data

Suggestive practicals include:

1. Apply data cleaning techniques on any dataset (eg. wine dataset). Techniques may include handling missing values, outliers, inconsistent values. A set of validation rules can be prepared based on the dataset and validations can be performed.
2. Apply data pre-processing techniques such as standardization/normalization, transformation, aggregation, discretization/binarization, sampling etc. on any dataset
3. Use Simple K-means algorithm for clustering on any dataset. Compare the performance of clusters by changing the parameters involved in the algorithm. Plot Mean Squared Error computed after each iteration using a line plot for any set of parameters

4. Apply Partitioning Methods, Hierarchical Methods, Density-Based Methods for clustering on a data set and compare the performance of the obtained results using different metrics
5. Create an ensemble using Random Forest and show the impact of bagging and boosting on the performance
6. Use Naive bayes, K-nearest, and Decision tree classification algorithms and build classifiers on any two datasets. Divide the data set into training and test set. Compare the accuracy of the different classifiers under the following situations:
   I. a) Training set = 75% Test set = 25% b) Training set = 66.6% (2/3rd of total), Test set = 33.3%
   II. Training set should be chosen by i) hold out method ii) Random subsampling iii) Cross-Validation. Compare the accuracy of the classifiers obtained.
   III. Data should be scaled to the standard format.

Project

Students should be promoted to take up one project on any UCI/kaggle/data.gov.in or on a dataset verified by the teacher. Preprocessing steps and at least one data mining technique should be shown on the selected dataset. This will allow the students to have practical knowledge of how to apply the various skills learned in the subject to a single problem/project.

DISCIPLINE SPECIFIC Elective (DSE 03b): Web Design and Development

Semester 5

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 03b Web Design and Development	**4**	**3**	**0**	**1**	Pass in Class XII	Knowledge of Structured Query Language (SQL)

Learning Objectives

The course aims at introducing the basic concepts and techniques of complete website-based programming. The student shall be able to develop simple, interactive, and dynamic websites using HTML, Javascript, PHP and database.

Learning outcomes

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

1. Build interactive and dynamic websites.
2. Use the client-side validation techniques using Javascript.
3. Write the server-side programming techniques with PHP for accessing the contents to/from the server.
4. Use GET and POST methods for exchanging data between client and server.
5. Learn to connect PHP with databases, save and retrieve data dynamically.

SYLLABUS OF DSE 03b

Unit 1. Introduction (2 hours)

Introduction to internet and web design. Basic concepts of web architecture.

Unit 2. HTML (11 hours)

Introduction to hypertext mark-up language (HTML), creating web pages, lists, elements of HTML, hyperlinks, tables, forms, inserting images.

Unit 3. Basics of Javascript (12 hours)

Document object model, data types and variables, functions, methods, and events, controlling program flow, client-side form validation.

Unit 4. Introduction to PHP (8 hours)

Basic syntax, defining variables and constants, data types, operators and expression, decision making statements, loop constructs, functions.

Unit 5. Handling HTML Form with PHP (6 hours)

Connecting an HTML form with PHP, submitting data to the server using GET and POST methods, GET vs POST methods.

Unit 6. Database Connectivity (6 hours)

Connectivity with database, database creation, creating tables, insertion and retrieval of the data from the database, data manipulation.

Essential/recommended readings

Suggested References

1. Nixon, R. **Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5,** 2018, O'Rielly.
2. Powell, T.A. (2017). **HTML & CSS: The Complete Reference**. 5th edition, 2017, Tata McGrawHill.
3. Duckett, J. **JavaScript and JQuery: Interactive Front-End Web Development**, 2014, Wiley.
4. Murach J., **Murach's PHP and MySQL,** 2nd Edition, 2014, Mike Murach & Associates.
5. Holzner S. **PHP: The Complete Reference,** 2017, McGraw Hill**.**

6. Ivan Bayross, **Web Enabled Commercial Application Development Using Html, Dhtml, Javascript, Perl CGI**, 2010, BPB Publications.

Suggested Practical List (If any): (30 Hours)

HTML

1. Create an HTML document with following formatting – Bold, Italics, Underline, Colors, Headings, Title, Font and Font Width, Background, Paragraph, Line Brakes, Horizontal Line, Marquee text.
2. Create an HTML document with Ordered and Unordered lists, Images, Internal and External linking.
3. Create an HTML document for displaying the current semester's timetable.
4. Create a student registration form using HTML which has the following controls:
    a. Text Box

b. Dropdown box
c. Option/radio button
d. Check boxes
e. Reset and Submit button.

Javascript

1. Write event-driven programs in JavaScript for the following:
    a. Enter a number and on click of a button print its factors.
    b. Print the smallest of five numbers entered by the user.
    c. Find the factorial of a number entered by the user.
    d. Take a number in an input text box, on click of a button, display its multiplication table.
2. Create a student registration form. Create functions to perform the following checks:
    a. Student id is a 10-digit alphanumeric value
    b. Name should be an alphabetical value (String)
    c. Non-empty fields like Age
3. Create a form containing various HTML elements and perform appropriate validations on each of them while submitting the form.

PHP

1. Write a PHP script to print the sum of even digits of a number.
2. Write a script in PHP to display a Multiplication Table.
3. Design a Student Registration form, using appropriate input fields consisting of following:
    a. Roll Number
    b. First Name
    c. Last Name
    d. Gender
    e. Department
    f. DOB

    Submit, retrieve the form data using the $_POST variable and display it.
4. Write PHP Code to create a database, connect to it, create tables, insert and access their contents.
5. Write PHP code to insert, delete, and retrieve data from the database. Create proper forms for performing the above operations. Display the messages such as “The record is added in the database!” when data is inserted into the database, “A record is deleted from the database” when data is deleted from the database. Use appropriate button names such as Add Data, Delete Data, and Display Data.

DISCIPLINE SPECIFIC Elective - (DSE-4a) : Applied Network Analytics

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Semester 6

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 04a Applied Network Analytics	**4**	**3**	**0**	**1**	Pass in Class XII	Knowledge of Python Programming

Learning Objectives

The course introduces basic concepts and methods analysis of networks whose foundation is graph theory. Distinction between a graph as an abstract structure and a real-life situation modeled as a network is detailed in this paper. The objective of this course is to expose the students to the strengths and capabilities of network analysis and their applications in real life. The students will be encouraged to apply the concepts taught in the course to real life problems using open-source software.

Learning outcomes

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

- Mapping of real world situation into networks
- Identify and apply quantitative network measures to characterize social networks at the local and global level
- Generate synthetic networks that satisfy properties of real world networks and its traversal
- Discover, analyze and evaluate the disjoint community structure of networks
- Model an information diffusion process for predictive analysis of social networks

SYLLABUS OF DSE 04a

Unit 1 (8 Hours)

Introduction to Network Science: Elementary Graph theory, Degree and degree distribution, Applications of network science in social network analysis, Network types.

Unit 2 (6 Hours)

Restructuring Data for use in Graph: Mapping of real world situation into networks, Representation of graphs, Transforming tabular and semi-structured data into graphs

Unit 3 (10 Hours)

Graph Traversal: Traversing the graph, Path, Distance, Path length, Shortest Path algorithms, Diameter, Density

Generative Network Models: Random, Small-world and Scale-free networks, Properties of real-world networks.

Unit 4 (10 Hours)

Vertex Importance and Centrality: Centrality measures, Hubs and authority, Page rank algorithm, Assortativity, Transitivity and Reciprocity, Calculating and illustrating vertex centrality

Unit 5 (11 Hours)

Community Structure: Different types of communities, Modularity, Algorithm for disjoint community detection

Information Diffusion in Social Networks: Cascading of information and innovations, Standard epidemic models for information diffusion

Essential/recommended readings

1. Chakraborty T. *Social Network Analysis*, 1st edition, Wiley India Pvt. Ltd., 2021.
2. McNulty, Keith. *Handbook of Graphs and Networks in People Analytics With Examples in R and Python.* 1st edition, CRC Press, 2022.

3. Yang S., Keller F. B., Zheng L.. *Social Network Analysis: Methods and Examples.* 1st edition, Sage Publications, 2017.

Additional References

1. Barabási A. L. , Pósfai M. *Network Science*, 1st edition, Cambridge University Press, 2016.
2. Easley, Kleinberg J. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*, 1st edition, Cambridge University Press, 2012.

Suggested Practical List (If any): (30 Hours)

Python Packages like igraph, NetworkX, NDlib etc. may be used for programming

1. Create a graph from an edge-list and another graph from an adjacency matrix. Explore the graph to determine its number of vertices and edges.
2. Plot a weighted network such that node size and edge width is proportional to their degree and edge weight respectively. Experiment with different layouts for visualization.
3. Compute and plot degree distribution of a small real-world network. Use appropriate functions to determine shortest paths between all pairs of vertices.
4. Generate three networks of 1000 nodes each using Random Network Model, Small World Network Model, Scale Free Model and compare their characteristics.
5. Compute different centrality measures to identify top-N nodes and compare their ranks with those obtained by PageRank method.
6. Apply community detection algorithms on a small real-world network (e.g. Karate club) and compare modularity using bar plot. Also plot the communities revealed with different colors.
7. Simulate diffusion trends for different epidemic models and present results using appropriate visuals.

DISCIPLINE SPECIFIC Elective - (DSE-4b) : Foundations of Computer Graphics

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Semester 6

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 04b Foundations of Computer Graphics	**4**	**3**	**0**	**1**	Pass in Class XII	**Programming in C/C++, Basic Matrices and Differential Calculus**

Learning Objectives

This course lays the foundation of Computer Graphics by introducing the fundamental concepts of modelling, rendering and interaction. The course focuses on rasterization technique for discussing the drawing and rendering of primitives. The course gives both the theoretical and practical knowledge to design, use, and understand computer graphics systems.

Learning outcomes

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

1. Describe Standard raster and vector scan devices as well as Graphical Input and output devices
2. Present the different stages of the traditional graphics pipeline in detail.
3. Implement algorithms for drawing basic primitives such as line and circle.
4. Implement algorithms for line clipping and polygon clipping.
5. Explain how models can be translated, scaled, rotated, and sheared by transformation matrices including the use of homogeneous coordinates and concatenated transformations
6. Understand and explain the basic principles of visible surface determination while rendering a scene.
7. Describe the process to compute the intensity at a particular point in the scene and use the same to shade the entire surface.
8. Understand the basics of simple computer animation.

SYLLABUS OF DSE 04b

Unit 1 **(5 hours)**

Introduction: Introduction to Graphics systems, Basic elements of Computer graphics, Applications of computer graphics, Architecture of Raster and Vector scan display devices, Color Lookup Table, Display devices (Cathode Ray Tube (CRT) , Colored CRTs, Direct View Storage tube(DVST), Plasma Panel, LCD, LED, Emissive and Non-emissive displays), Input devices.

Unit 2 (8 hours)

Drawing and clipping primitives: Raster scan line (Digital Differential Analyzer (DDA) and Bresenhams) and circle drawing algorithms, line clipping using Cohen and Sutherland Line clipping algorithm and polygon clipping using Sutherland and Hodgeman Polygon clipping algorithm.

Unit 3 (15 hours)

Transformation and Viewing: Basic 2D transformations (Translation, Rotation, scaling, reflection and shearing), Homogenous coordinates, composite transformation, 3D Geometric Transformations, Viewing Transformations (Projections- Parallel and Perspective), Vanishing points.

Unit 4 (5 hours)

Geometric Modeling: Polygon Mesh Representation, Cubic Polynomial curves (Hermite).

Unit 5 (8 hours)

Visible Surface determination/Hidden Surface removal and Surface Rendering: Need for hidden surface removal, Z-buffer algorithm and area subdivision algorithm for visible surface determination. Phong Illumination model, Phong and Gouraud shading models, Halftoning and Dithering.

Unit 6 (4 hours)

Basics of Computer Animation: Storyboard layout, keyframe systems, simulating motion, morphing.

Essential/recommended readings

1. Hearn, D & Baker, M.P. Computer Graphics. 2nd edition. Prentice Hall of India, 2009.
2. Foley, J. D., Dam, A.V, Feiner, S. K., & Hughes, J. F. . Computer Graphics: Principles and Practice in C. 2nd edition. Pearson Education, 2002.
3. Rogers, D. F. . Mathematical Elements for Computer Graphics. 2nd edition. McGraw Hill. 2017.

Additional References:

1. Bhattacharya, S. Computer Graphics. Oxford University Press, 2018.
2. Marschner, S., & Shirley, P. Fundamentals of Computer Graphics. 5th edition. CRC Press, 2021.

Online references/material:

Suggested Practical List (If any): (30 Hours)

1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous Coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw the Hermite curve.