Savitribai Phule Pune University
(Formerly University of Pune)
DEPARTMENT OF TECHNOLOGY

Board of Studies
B. Tech Data Science
Curriculum Structure
Applicable from Academic Year 2024-25

Intake: 60
Duration: 4 Year

Eligibility:
1st Year: 12th Science with Maths
Direct 2nd Year: Diploma/ Polytechnic in Any Branch

I. PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

Graduates can:

PEO1. Utilize their proficiencies in the fundamental knowledge of basic sciences, mathematics, Artificial Intelligence, data science and statistics to build systems that require management and analysis of large volumes of data.
PEO2. Advance their technical skills to pursue pioneering research in the field of AI and Data Science and create disruptive and sustainable solutions for the welfare of ecosystems.
PEO3. Think logically, pursue lifelong learning and collaborate with an ethical attitude in a multidisciplinary team.
PEO4. Design and model AI based solutions to critical problem domains in the real world.
PEO5. Exhibit innovative thoughts and creative ideas for effective contribution towards economy building.

II. PROGRAM OUTCOMES (POs)

PO# Graduate Attribute

PO1 Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2 Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3 Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4 Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5 Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6 The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7 Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9 Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10 Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11 Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12 Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

III. PROGRAM SPECIFIC OUTCOMES (PSOs)

Graduates should be able to:

PSO1 Efficient AI-based Processes: Evolve AI-based efficient domain-specific processes for effective decision making in several domains such as business and governance domains.

PSO2 Actionable Insights: Arrive at actionable foresight, insight, and hindsight from data for solving business and engineering problems.

PSO3 Theoretical and Practical Knowledge: Create, select, and apply the theoretical knowledge of AI and Data Analytics along with practical industrial tools and techniques to manage and solve wicked societal problems.

PSO4 Data Analytics and Visualization: Develop data analytics and data visualization skills, skills pertaining to knowledge acquisition, knowledge representation, and knowledge engineering, and hence be capable of coordinating complex projects.

PSO5 Fundamental Research: Able to carry out fundamental research to cater the critical needs of the society through cutting-edge technologies of AI.

I[st] YEAR SEMESTER-1

S. No.	Course Code	Course	L	T	P	Credits
1.	BTDS 101	Linear Algebra and Calculus	3	1	0	4
2.	BTDS 102	Elements of Data Science Engineering	3	0	0	3
3.	BTDS 103	Object Oriented Programming Language	3	0	0	3
4.	BTDS 104	English for Skill Enhancement	2	0	2	4
5.	BTDS 105	IT Workshop	0	0	2	2
6.	BTDS 103L	Object Oriented Programming Language Lab	0	0	4	4
7.	BTDS 106	Seminar-I	0	0	2	2
		Induction Program				
		Total	**11**	**1**	**10**	**22**

I[st] YEAR SEMESTER-2

S. No.	Course Code	Course	L	T	P	Credits
1.	BTDS 201	Applied Statistics*	3	1	0	4
2.	BTDS 202	Programming for Problem Solving *	3	1	0	4
3.	BTDS 203	Data Security in Data Science	3	0	1	4
4.	BTDS 204	Software Engineering	2	0	0	2
5.	BTDS 201L	Applied Statistics Lab	0	0	2	2
6.	BTDS 202L	Programming for Problem Solving Laboratory	0	0	4	4
7.	BTDS 205	Seminar-II	0	0	2	2
		Total	**11**	**2**	**9**	**22**

***L:** Lecture, **T:** Tutorial, **P:** Practical, **Cr:** Credits*
***ETE:** End Term Exam, **IA:** Internal Assessment*

Faculty can assign the topics for Seminar-I and Seminar-II based on recent current related topics.

IInd YEAR SEMESTER-3

S. No.	Course Code	Course Title	L	T	P	Credits
1	BTDS 301	Mathematics for Data Science	3	1	0	4
2	BTDS 302	Introduction to Data Analytics*	3	0	0	3
3	BTDS 303	Advance Data Structures	3	0	0	3
4	BTDS 304	Web Framework	3	0	0	3
5	BTDS 305	Computer Organization and Architecture	3	0	0	3
6	BTDS 303L	Advance Data Structures Lab	0	0	2	2
7	BTDS 304L	Web Framework Lab	0	0	2	2
8	BTDS 306L	Data Visualization- R Programming/ Power BI	0	0	4	2
		Total	**15**	**1**	**8**	**22**

IInd YEAR SEMESTER-4

S. No.	Course Code	Course Title	L	T	P	Credits
1	BTDS 401	Discrete Mathematics	3	0	0	3
2	BTDS 402	Introduction to Artificial Intelligence*	3	0	0	3
3	BTDS 403	Database Management Systems	3	0	0	3
4	BTDS 404	Operating Systems	3	0	0	3
5	BTDS 403L	Database Management Systems Lab	0	0	2	1
6	BTDS 404L	Operating Systems Lab	0	0	2	1
7	BTDS 405L	Node JS/ React JS	0	0	4	4
8	BTDS 406L	Real-time Research Project/Field Based Research Project including Seminar	0	0	4	4
		Total	**12**	**0**	**12**	**22**

III[rd] YEAR SEMESTER-5

S. No.	Course Code	Course Title	L	T	P	Credits
1	BTDS 501	Design and Analysis of Algorithms	3	0	0	3
2	BTDS 502	R Programming	3	0	0	3
3	BTDS 503	Business Economics & Financial Analysis	2	0	0	2
4	BTDS 504	Descriptive Analytics*	3	0	0	3
5		Professional Elective-I	2	0	1	3
6	BTDS 502L	R Programming Lab	0	0	2	2
7	BTDS 503L	Descriptive Analytics Lab	0	0	2	2
8	BTDS 505L	ETL-Kafka/Talend	0	0	4	4
		Total	**13**	**0**	**9**	**22**

III[rd] YEAR SEMESTER-6

S. No.	Course Code	Course Title	L	T	P	Credits
1	BTDS 601	Automata Theory and Compiler Design	3	0	0	3
2	BTDS 602	Machine Learning*	3	0	0	3
3	BTDS 603	Predictive Analytics*	3	0	0	3
4	BTDS 604	Professional Practice, Law & Ethics	2	0	0	2
5		Professional Elective – II	2	0	1	3
6	BTDS 602L	Machine Learning Lab	0	0	2	2
7	BTDS 603L	Predictive Analytics Lab	0	0	2	2
8	BTDS 605L	Industrial Oriented Mini Project/ Internship/Skill Development Course (UI design-Flutter)	0	2	2	4
		Total	**13**	**2**	**7**	**22**

IVth YEAR SEMESTER-7

S. No.	Course Code	Course Title	L	T	P	Credits
1	BTDS 701	Big Data Analytics*	3	0	0	3
2	BTDS 702	Web and Social Media Analytics	3	0	0	3
3		Professional Elective -III	3	0	0	3
4		Professional Elective -IV	3	0	0	3
5		Open Elective - I	2	0	1	3
6		Professional Elective -III Lab	0	0	2	2
7	BTDS 701L	Big Data Analytics Lab	0	0	2	2
8	BTDS 704	Project Stage – I including Seminar	0	0	3	3
		Total	**14**	**0**	**11**	**22**

IVth YEAR SEMESTER-8

S. No.	Course Code	Course Title	L	T	P	Credits
1		Professional Elective - V	3	0	1	4
2		Professional Elective – VI	3	0	1	4
3		Open Elective – II	3	0	0	3
4		Project Stage – II including Seminar*	0	0	22	11
		Total	**9**	**0**	**22**	**22**
		Total Credits			**176**	

***MC – Satisfactory/Unsatisfactory, #Skill Course - 1 credit with 2 Practical Hours**

Professional Elective-I

BTDS 511PE	Graph Theory
BTDS 512PE	Advanced Computer Architecture
BTDS 513PE	Web Programming
BTDS 514PE	Image Processing
BTDS 515PE	Computer Graphics

Professional Elective - II

BTDS 621PE	Software Testing Methodologies
BTDS 622PE	Information Retrieval Systems
BTDS 623PE	Pattern Recognition
BTDS 624PE	Computer Vision and Robotics
BTDS 625PE	Data Warehousing and Business Intelligence

Professional Elective - III

BTDS 731PE	Internet of Things
BTDS 732PE	Data Mining
BTDS 733PE	Scripting Languages
BTDS 734PE	Mobile Application Development

[#] **Courses in PE - III and PE - III Lab must be in 1-1 correspondence.**

Professional Elective -IV

BTDS 741PE	Quantum Computing
BTDS 742PE	Expert Systems
BTDS 743PE	Cloud Computing
BTDS 744PE	Game Theory
BTDS 745PE	Knowledge Representation and Reasoning

Professional Elective - V

BTDS 851PE	Social Network Analysis
BTDS 852PE	Federated Machine Learning
BTDS 853PE	Augmented Reality & Virtual Reality
BTDS 854PE	Web Security
BTDS 855PE	Ad-hoc & Sensor Networks

Professional Elective – VI

BTDS 861PE	Speech and Video Processing
BTDS 862PE	Robotic Process Automation
BTDS 863PE	Randomized Algorithms
BTDS 864PE	Cognitive Computing
BTDS 865PE	Semantic Web

Open Elective I:

1. BTDS 721OE: Introduction to Natural Language Processing
2. BTDS 722OE: AI applications

Open Elective II:

1. BTDS 831OE: Chatbots
2. BTDS 832OE: Evolutionary Computing

Mapping of Course Outcome and Programme Outcome

Year	Semester	Course Name	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3	PEO 1	PEO 2	PEO 3	PEO 4	PEO 5
I	1	BTDS 101: Linear Algebra and Calculus	3	2	3	1	2	0	0	0	1	2	1	2	3	2	3	3	2	1	2	1
		BTDS 102: Elements of Data Science	3	3	2	3	2	1	0	1	2	3	2	2	3	3	2	3	2	3	2	2
		BTDS 103: Object Oriented Programming	3	3	3	2	3	1	0	0	2	3	1	2	3	2	3	3	2	2	3	2
		BTDS 104: English for Skill Enhancement	2	1	2	0	1	1	0	3	3	3	2	2	1	1	1	1	1	2	1	1
		BTDS 105: IT Workshop	3	2	3	3	3	1	1	1	3	3	2	3	3	3	2	3	2	3	3	2
		BTDS 106: Seminar-I	2	1	2	2	1	1	0	2	3	3	2	2	1	2	1	2	2	3	1	1
	2	BTDS 201: Applied Statistics	3	3	2	2	2	0	0	0	0	0	0	0	3	3	2	3	3	3	3	2
		BTDS 202: Programming for Problem Solving	3	3	3	2	3	0	0	0	0	0	0	0	3	3	3	3	3	3	3	2
		BTDS 203: Data Security in Data	3	3	2	2	2	0	0	0	0	0	0	0	3	3	3	3	3	3	3	2

		Science																				
		BTDS 204: Software Engineering	3	3	2	3	2	0	0	0	0	0	0	0	3	3	2	3	3	3	3	2
		BTDS 201L: Applied Statistics Lab	3	3	2	3	2	0	0	0	0	0	0	0	3	3	3	3	3	3	3	2
		BTDS 202L: Programming for Problem Solving Lab	3	3	3	2	2	0	0	0	0	0	0	0	3	3	3	3	3	3	3	2
		BTDS 205: Seminar-II	2	3	2	3	2	0	0	0	0	0	0	0	3	3	2	3	3	3	3	2
II	3	**BTDS 204 Data Structures & Algorithms**	3	3	2	3	2	2	1	-	-	-	-	1	3	2	2	2	2	2	3	3
		BTDS 205 Database Management Systems	3	3	2	2	3	2	1	-	-	-	-	1	3	3	2	2	2	2	3	3
		BTDS 304 Computer Networks	3	3	2	3	3	2	1	-	-	-	-	1	3	2	3	2	2	2	3	3
		BTDS 205 Operating Systems	3	3	3	3	3	2	1	-	-	-	-	2	3	2	3	2	2	2	3	3
		BTDS 201 Discrete Mathematics	3	3	2	2	2	2	1	-	-	-	-	1	3	3	2	2	2	2	3	3

		BTDS 202 Object - Oriented Programming	3	3	2	3	2	2	1	-	-	-	-	1	3	2	2	2	2	2	3	3
	4	**BTDS 302 Data Mining**	3	3	3	2	2	2	2	3	2	2	3	2	3	3	2	3	2	3	2	2
		BTDS 303 Operating Systems	3	3	3	2	2	3	2	2	3	2	2	3	3	2	3	2	3	3	2	2
		BTDS 304 Design and Analysis of Algorithms	3	3	3	2	3	2	3	2	3	2	3	2	3	3	3	2	3	3	2	
		BTDS 305 Data Visualization	3	3	3	2	2	2	3	2	3	2	2	3	3	3	2	3	3	2	3	2
		Design and Analysis of Algorithms	3	3	3	2	2	-	-	-	-	-	-	-	3	3	2	-	-	-	-	-
		Operating Systems	3	3	3	2	2	-	-	-	-	-	-	-	3	3	2	-	-	-	-	-
III	5	**Data Warehousing and Mining**	3	3	3	2	2	-	-	-	-	-	-	-	3	3	2	-	-	-	-	-
		Compiler Design	3	3	3	2	2	-	-	-	-	-	-	-	3	3	2	-	-	-	-	-
		Artificial Intelligence	3	3	3	2	2	-	-	-	-	-	-	-	3	3	2	-	-	-	-	-
	6	**Machine Learning**	3	3	2	2	3	-	-	-	-	-	-	-	3	3	3	3	3	3	3	3
		Data Visualization	3	3	1	2	3	-	-	-	-	-	-	-	3	3	3	3	3	3	3	3

		Cloud Computing	3	3	2	2	3	-	-	-	-	-	-	-	3	3	3	3	3	3	3	3
		Data Mining and Warehousing	3	3	1	3	3	-	-	-	-	-	-	-	3	3	3	3	3	3	3	3
		Operating Systems	3	3	2	2	3	-	-	-	-	-	-	-	3	3	3	3	3	3	3	3
		Big Data Analytics*	3	3	3	2	2	2	2	3	2	2	3	2	3	3	2	3	2	3	2	2
		Web and Social Media Analytics	3	3	3	2	2	3	2	2	3	2	2	3	3	2	3	2	3	3	2	2
	7	**Professional Elective -III**	3	3	3	2	3	2	3	2	3	2	3	2	3	3	3	2	3	3	2	
		Professional Elective -IV	3	3	3	2	2	2	3	2	3	2	2	3	3	3	2	3	3	2	3	2
IV		**Open Elective - I**	3	3	3	2	2	2	2	3	2	2	3	2	3	3	2	3	2	3	2	2
		Professional Elective - V	3	3	3	2	2	3	2	2	3	2	2	3	3	2	3	2	3	3	2	2
		Professional Elective – VI	3	3	3	2	3	2	3	2	3	2	3	2	3	3	3	2	3	3	2	
	8	**Open Elective – II**	3	3	3	2	2	2	3	2	3	2	2	3	3	3	2	3	3	2	3	2
		Project Stage – II including Seminar*	3	3	3	2	3	2	3	2	3	2	3	2	3	3	3	2	3	3	2	

BTDS 101: LINEAR ALGEBRA AND CALCULUS

B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic understanding of high school mathematics (algebra, geometry, trigonometry).

Course Objective:

1. Understand fundamental concepts of linear algebra and calculus.
2. Develop problem-solving skills using algebraic and calculus methods.
3. Apply linear algebra and calculus techniques to real-world problems.
4. Integrate knowledge of linear algebra and calculus in advanced topics in data science and engineering.

Learning Course Outcome:

After successful completion of the course, students will able to:

1. Demonstrate a comprehensive understanding of vector and matrix operations, including addition, multiplication, and finding determinants and inverses.
2. Solve systems of linear equations using Gaussian elimination and Gauss-Jordan elimination, and determine the nature of the solution sets.
3. Calculate eigenvalues and eigenvectors of matrices, and understand their applications in data science and engineering contexts.
4. Understand and apply the concepts of functions, limits, and continuity in mathematical and real-world problems.
5. Compute derivatives of functions using various differentiation rules, and apply differentiation techniques to optimization and related rates problems.
6. Calculate indefinite and definite integrals using various techniques of integration, and apply integration to problems involving areas and volumes.

Course Contents

UNIT-I	Vectors and Matrices	07 Hours

Introduction to vectors: definitions, operations, and properties, Dot product and cross product, Matrices: definitions, types, and operations, Matrix multiplication and properties, Determinants and their properties, Inverse of a matrix. **Hands-on Exercises:**

- Vector operations and visualizations
- Matrix multiplication and determinant calculations

UNIT-II	Systems of Linear Equations	08 Hours

Introduction to systems of linear equations, Gaussian elimination and Gauss-Jordan elimination, Row echelon form and reduced row echelon form, Solution sets: unique, infinite, and no solution, Applications in engineering and data science. **Hands-on Exercises:**

- Solving systems of linear equations
- Applications in real-world scenarios

UNIT-III	Eigenvalues and Eigenvectors	08 Hours
Introduction to eigenvalues and eigenvectors, Characteristic polynomial and finding eigenvalues, finding eigenvectors corresponding to eigenvalues, Diagonalization of matrices, Applications in data science and engineering. **Hands-on Exercises:** • Finding eigenvalues and eigenvectors • Diagonalizing matrices and applications		
UNIT-IV	**Functions and Limits**	**07 Hours**
Introduction to functions: definitions, types, and properties, Domain and range of functions, Composition of functions and inverse functions, Limits: definitions, properties, and calculation techniques, Continuity and its implications. **Hands-on Exercises:** • Function operations and compositions • Calculating limits and exploring continuity		
UNIT-V	**Differentiation**	**08 Hours**
Definition of the derivative and interpretation, Basic differentiation rules (power, product, quotient, chain), Higher-order derivatives, Implicit differentiation, Applications: optimization, motion, related rates. **Hands-on Exercises:** • Differentiation practice problems • Real-world applications of derivatives		
UNIT-VI	**Integration**	**08 Hours**
Introduction to integration: definitions and properties, Indefinite and definite integrals, Fundamental Theorem of Calculus, Techniques of integration: substitution, integration by parts, Applications: area under curves, volume of solids of revolution. **Hands-on Exercises:** • Integration practice problems • Applications of integration in various fields		

REFERENCE BOOK:

1. "Linear Algebra and Its Applications" by Gilbert Strang
2. "Introduction to Linear Algebra" by Gilbert Strang
3. "Calculus: Early Transcendentals" by James Stewart
4. "Calculus" by Michael Spivak

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4
CO1	3	2	-	-	-	3	-	-	2
CO2	3	3	-	2	-	3	-	-	2
CO3	3	2	-	-	3	3	-	2	2
CO4	3	2	2	-	-	-	3	-	2
CO5	3	3	2	-	2	-	2	2	-
CO6	3	3	2	-	2	-	3	-	2
Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 102: ELEMENTS OF DATA SCIENCE ENGINEERING
B.Tech. I Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: 10 + 2 Physics

Course Objective:

1. **Introduce Fundamental Concepts**: Provide a comprehensive understanding of the basic concepts and principles of data science.
2. **Explore the Data Lifecycle**: Familiarize students with the stages of the data lifecycle, including data collection, cleaning, analysis, modeling, and visualization.
3. **Hands-on Experience with Tools**: Equip students with practical skills using essential data science tools and programming libraries.
4. **Develop Analytical Skills**: Enhance students' ability to perform exploratory data analysis and interpret data insights.
5. **Understand Data Modeling**: Introduce machine learning concepts and algorithms, and their application in building predictive models.
6. **Communicate Insights Effectively**: Teach students to create effective data visualizations and communicate findings clearly to diverse audiences.

Learning Course Outcome:
After successful completion of the course, students will able to:

CO1 Understand and Define Data Science: Articulate the scope, significance, and applications of data science across various industries.

CO2 Navigate the Data Lifecycle: Describe and implement the stages of the data lifecycle from data collection to deployment.

CO3 Collect and Clean Data: Apply techniques for data collection from various sources and perform data cleaning to prepare datasets for analysis.

CO4 Perform Exploratory Data Analysis: Use statistical methods and visualization techniques to explore and analyze data.

CO5 Build and Evaluate Models: Implement basic machine learning algorithms, evaluate their performance, and interpret the results.

CO6 Visualize Data Effectively: Create clear and impactful data visualizations using advanced tools and techniques.

CO7 Communicate Data Findings: Present data insights effectively, using storytelling techniques to convey complex information to non-technical audiences.

CO8 Apply Tools and Techniques: Demonstrate proficiency in using Python libraries (e.g., NumPy, Pandas, Matplotlib, scikit-learn) and data visualization tools (e.g., Tableau, Power BI).

Course Contents

UNIT-I	Introduction to Data Science	07 Hours

- **Definition and Scope**
  - What is Data Science?
  - Evolution and significance of Data Science
  - Key components and interdisciplinary nature
- **Applications of Data Science**
  - Real-world applications in various industries (e.g., healthcare, finance, marketing)
  - Case studies of successful data science projects

UNIT-II	Data Acquisition and Collection	08 Hours

- **Data Sources**: Identifying and integrating various data sources such as databases, APIs, web scraping, IoT devices, and public datasets.
- **Data Collection Tools**: Tools and technologies for data collection such as Python libraries (e.g., BeautifulSoup, Scrapy), and ETL (Extract, Transform, Load) tools like Apache NiFi, Talend, and Informatica.
- **APIs and Web Scraping**: Techniques to gather data from the web and APIs.

UNIT-III	Data Storage and Management	08 Hours

- **Databases**: Relational databases (e.g., MySQL, PostgreSQL) and NoSQL databases (e.g., MongoDB, Cassandra) for structured and unstructured data storage.
- **Data Warehousing**: Systems designed for query and analysis, such as Amazon Redshift, Google BigQuery, and Snowflake.
- **Data Lakes**: Storage repositories that hold large amounts of raw data in its native format, such as Hadoop HDFS and Amazon S3.

UNIT-IV	Data Processing and Cleaning	07 Hours

- **Data Cleaning**: Techniques for handling missing values, outliers, and duplicates using libraries like pandas and dplyr.
- **Data Transformation**: Methods for data normalization, encoding, and scaling using tools such as Python's pandas and scikit-learn.
- **Batch and Stream Processing**: Frameworks for large-scale data processing, like Apache Spark for batch processing and Apache Kafka for stream processing.

UNIT-V	Data Exploration and Visualization	08 Hours

- **Exploratory Data Analysis (EDA):** Techniques for summarizing and visualizing data distributions and relationships using tools like Jupyter notebooks, pandas, Matplotlib, Seaborn, and Plotly.
- **Data Visualization**: Creating interactive and static visualizations using libraries such as Tableau, Power BI, D3.js, and Dash.

UNIT-VI	Cloud Computing and Deployment	07 Hours

- **Cloud Platforms**: Services like AWS, Google Cloud Platform (GCP), and Microsoft Azure for hosting and managing data science applications.
- **Containerization and Orchestration**: Tools like Docker and Kubernetes for deploying and managing microservices and data science models in a scalable manner.
- **Model Deployment**: Techniques for deploying machine learning models as APIs using frameworks like Flask, FastAPI, TensorFlow Serving, and AWS SageMaker.

Course Activities and Assessments

- Hands-on Labs: Practical exercises and lab sessions to apply theoretical concepts.
- Mini Projects: Small-scale projects involving real-world datasets to reinforce learning.
- Quizzes and Exams: Regular assessments to test understanding of key concepts.
- Group Discussions: Collaborative discussions and presentations on data science topics.

References
Textbooks

1. **"Python for Data Analysis" by Wes McKinney**
   A practical guide to data manipulation and analysis using Python and its powerful libraries like Pandas, NumPy, and IPython.
2. **"Data Science from Scratch: First Principles with Python" by Joel Grus**
   A hands-on guide to understanding and implementing data science algorithms from scratch using Python.
3. **"Introduction to Machine Learning with Python: A Guide for Data Scientists" by Andreas C. Müller and Sarah Guido**
   A comprehensive introduction to machine learning, covering the fundamentals and practical applications using Python.
4. **"Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei**
   A detailed textbook on data mining techniques, covering both theory and practical applications.
5. **"The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling" by Ralph Kimball and Margy Ross**
   A foundational book on data warehousing and dimensional modeling.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

COs/POs & PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	-	-	1	-	-	3	2	1	-	-
CO2	3	3	2	2	2	-	-	3	3	3	2	-
CO3	3	3	-	-	3	-	-	3	3	3	2	-
CO4	3	3	1	3	3	-	-	3	3	2	2	-
CO5	3	3	1	3	3	-	-	3	3	3	2	-
CO6	2	3	-	2	3	-	-	3	3	3	3	-
CO7	2	3	2	-	3	-	-	3	3	3	3	-
CO8	3	3	1	2	3	-	-	3	3	3	3	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 103: OBJECT ORIENTED PROGRAMMING LANGUAGE

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic understanding of programming concepts.

Course Objective:

1. To introduce the fundamental concepts of object-oriented programming.
2. To provide a comprehensive understanding of the differences and similarities between Java and C++.
3. To develop problem-solving skills using object-oriented principles.
4. To apply object-oriented concepts in software development.
5. To understand the principles of inheritance, polymorphism, encapsulation, and abstraction.
6. To familiarize students with standard libraries and frameworks in Java and C++.

Learning Course Outcome:
After successful completion of the course, students will able to:

CO1: Understand and apply the basic principles of object-oriented programming in Java and C++.
CO2: Develop and use classes and objects in Java and C++.
CO3: Implement inheritance to promote code reuse and develop a hierarchical class structure.
CO4: Use polymorphism to create flexible and reusable code.
CO5: Apply encapsulation and abstraction to design robust software.
CO6: Handle exceptions and perform file I/O operations in Java and C++.
CO7: Utilize advanced features such as templates, generics, and multithreading.

Course Contents

UNIT	Title	Hours
UNIT-I	**Introduction to Object-Oriented Programming (OOP)**	**07 Hours**
Procedural vs. Object-Oriented Programming, Basic Concepts: Classes, Objects, Methods, and Messages, Benefits of OOP. Principles of Object-Oriented Programming, Benefits of OOP, Introduction to Java and C++, Basic Syntax and Structure, Compiling and Running Programs.		
UNIT-II	**Classes and Objects**	**08 Hours**
Defining Classes and Creating Objects, Constructors and Destructors, Access Specifiers, Static Members and Methods, This Pointer in C++ / This Keyword in Java.		
UNIT-III	**Inheritance**	**08 Hours**
Concept of Inheritance, Types of Inheritance: Single, Multiple (C++), Multilevel, Hierarchical, Hybrid, Constructor and Destructor Calls in Inheritance, Method Overriding, The super Keyword in Java, Virtual Base Classes in C++		
UNIT-IV	**Polymorphism**	**07 Hours**

Function Overloading and Operator Overloading in C++, Method Overloading in Java, Virtual Functions and Abstract Classes in C++, Interfaces and Abstract Classes in Java, Runtime Polymorphism, Type Casting.		
UNIT-V	**Encapsulation and Abstraction**	**08 Hours**
Data Hiding and Encapsulation, Access Modifiers and Their Usage, Abstract Classes and Methods, Packages in Java and Namespaces in C++, Implementing Abstraction		
UNIT-VI	**Advanced Object-Oriented Concepts & Real-World Applications and Case Studies**	**08 Hours**
Exception Handling in Java and C++, File Handling: Stream Classes in C++, File I/O in Java, Templates and Generics, Standard Template Library (STL) in C++, Collections Framework in Java, Multithreading and Concurrency: Thread Management in C++, Thread Class and Runnable Interface in Java Developing a Simple Application: Step-by-Step Approach, Case Study: Banking System Application, Case Study: Library Management System, Best Practices in OOP, Debugging and Testing Object-Oriented Programs		

References

TEXT BOOKS

1. Jeri R. Hanly and Elliot B.Koffman, Problem solving and Program Design in C 7th Edition, Pearson
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition)
3. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
4. E. Balagurusamy, Computer fundamentals and C, 2nd Edition, McGraw-Hill
5. Yashavant Kanetkar, Let Us C, 18th Edition, BPB
6. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression)
7. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
8. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition
9. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3	PSO4
CO1: Understand and apply the basic principles of object-oriented programming in Java and C++	3	3	2	2	3	-	-	2	2	2	2
CO2: Develop and use classes and objects in Java and C++	3	3	2	2	3	-	-	3	3	3	2
CO3: Implement inheritance to promote code reuse and develop a hierarchical class structure	3	3	2	2	3	-	-	3	3	3	2
CO4: Use polymorphism to create flexible and reusable code	3	3	2	2	3	-	-	3	3	3	2
CO5: Apply encapsulation and abstraction to design robust software	3	3	2	2	3	-	-	3	3	3	2
CO6: Handle exceptions and perform file I/O operations in Java and C++	3	3	2	2	3	-	-	3	3	3	2
CO7: Utilize advanced features such as templates, generics, and multithreading	3	3	2	2	3	-	-	3	3	3	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 104: ENGLISH FOR SKILL ENHANCEMENT
B.Tech. I Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: English Grammar basic knowledge

Course Objective:

1. Improve the language proficiency of students in English with an emphasis on Vocabulary, Grammar, Reading and Writing skills.
2. Develop study skills and communication skills in various professional situations.
3. Equip students to study engineering subjects more effectively and critically using the theoretical and practical components of the syllabus.

Learning Course Outcome:
After successful completion of the course, students will able to:

1. Understand the importance of vocabulary and sentence structures.
2. Choose appropriate vocabulary and sentence structures for their oral and written communication.
3. Demonstrate their understanding of the rules of functional grammar.
4. Develop comprehension skills from the known and unknown passages.
5. Take an active part in drafting paragraphs, letters, essays, abstracts, précis and reports in various contexts.
6. Acquire basic proficiency in reading and writing modules of English.

Course Contents

UNIT-I	Chapter entitled 'Toasted English' by R.K.Narayan from "English: Language, Context and Culture" published by Orient BlackSwan, Hyderabad.	**07 Hours**

Vocabulary: The Concept of Word Formation -The Use of Prefixes and Suffixes - Acquaintance with Prefixes and Suffixes from Foreign Languages to form Derivatives - Synonyms and Antonyms

Grammar: Identifying Common Errors in Writing with Reference to Articles and Prepositions.

Reading: Reading and Its Importance- Techniques for Effective Reading.

Writing: Sentence Structures -Use of Phrases and Clauses in Sentences- Importance of Proper Punctuation- Techniques for Writing precisely – Paragraph Writing –

	Types, Structures and Features of a Paragraph - Creating Coherence-Organizing Principles of Paragraphs in Documents.	
UNIT-II	Chapter entitled **'Appro JRD' by Sudha Murthy** from ***"English: Language, Context and Culture" published*** by Orient BlackSwan, Hyderabad.	**08 Hours**
Vocabulary: Words Often Misspelt - Homophones, Homonyms and Homographs **Grammar:** Identifying Common Errors in Writing with Reference to Noun-pronoun Agreement and Subject-verb Agreement. **Reading:** Sub-Skills of Reading – Skimming and Scanning – Exercises for Practice **Writing:** Nature and Style of Writing- Defining /Describing People, Objects, Places and Events — Classifying- Providing Examples or Evidence.		
UNIT-III	Chapter entitled **'Lessons from Online Learning' by F.Haider Alvi, Deborah Hurst et al** from ***"English: Language, Context and Culture"*** published by Orient BlackSwan, Hyderabad.	**08 Hours**
Vocabulary: Words Often Confused - Words from Foreign Languages and their Use in English. **Grammar:** Identifying Common Errors in Writing with Reference to Misplaced Modifiers and Tenses. **Reading:** Sub-Skills of Reading – Intensive Reading and Extensive Reading – Exercises for Practice. **Writing:** Format of a Formal Letter-Writing Formal Letters E.g.., Letter of Complaint, Letter of Requisition, Email Etiquette, Job Application with CV/Resume.		
UNIT-IV	Chapter entitled **'Art and Literature' by Abdul Kalam** from ***"English: Language, Context and Culture"*** published by Orient BlackSwan, Hyderabad.	**07 Hours**
Vocabulary: Standard Abbreviations in English **Grammar:** Redundancies and Clichés in Oral and Written Communication. **Reading**: Survey, Question, Read, Recite and Review (SQ3R Method) - Exercises for Practice **Writing:** Writing Practices- Essay Writing-Writing Introduction and Conclusion -Précis Writing.		
UNIT-V	Chapter entitled **'Go, Kiss the World' by Subroto Bagchi**	**08 Hours**

	from ***"English: Language, Context and Culture"*** published by Orient BlackSwan, Hyderabad.	

Vocabulary: Technical Vocabulary and their Usage
Grammar: Common Errors in English (*Covering all the other aspects of grammar which were not covered in the previous units*)
Reading: Reading Comprehension-Exercises for Practice
Writing: Technical Reports- Introduction – Characteristics of a Report – Categories of Reports Formats- Structure of Reports (Manuscript Format) -Types of Reports - Writing a Report.

- **Note: 1**. As the syllabus of English given in AICTE *Model Curriculum-2018 for B.Tech First Year is **Open-ended***, besides following the prescribed textbook, it is required to prepare teaching/learning materials **by the teachers collectively** in the form of handouts based on the needs of the students in their respective colleges for effective teaching/learning in the class.

- **Note**: **2**. Based on the recommendations of NEP2020, teachers are requested to be flexible to adopt Blended Learning in dealing with the course contents. They are advised to teach 40 percent of each topic from the syllabus in blended mode.

TEXT BOOK:

1. "English: Language, Context and Culture" by Orient Black Swan Pvt. Ltd, Hyderabad. 2022. Print.

REFERENCE BOOKS:

1. Effective Academic Writing by Liss and Davis (OUP)
2. Richards, Jack C. (2022) Interchange Series. Introduction, 1,2,3. Cambridge University Press
3. Wood, F.T. (2007). Remedial English Grammar. Macmillan.
4. Chaudhuri, Santanu Sinha. (2018). Learn English: A Fun Book of Functional Language, Grammar and Vocabulary. (2nd ed.,). Sage Publications India Pvt. Ltd.
5. (2019). Technical Communication. Wiley India Pvt. Ltd.
6. Vishwamohan, Aysha. (2013). English for Technical Communication for Engineering Students. Mc Graw-Hill Education India Pvt. Ltd.
7. Swan, Michael. (2016). Practical English Usage. Oxford University Press. Fourth Edition.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 105: IT WORKSHOP
B.Tech. I Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic Knowledge of computer

Course Objective:

The IT Workshop for engineers is a training lab course spread over 60 hours. The modules include training on PC Hardware, Internet & World Wide Web and Productivity tools including Word, Excel, PowerPoint and Publisher.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Perform Hardware troubleshooting
- Understand Hardware components and inter dependencies
- Safeguard computer systems from viruses/worms
- Document/ Presentation preparation
- Perform calculations using spreadsheets

Course Contents

PC Hardware	07 Hours

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Internet & World Wide Web

Task1: **Orientation & Connectivity Boot Camp:** Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: **Search Engines & Netiquette:** Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD

Task 1 – Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office or equivalent (FOSS) tool word: Importance of LaTeX and MS office or equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using LaTeX and word — Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Task 2: Using LaTeX and Word to create a project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 3: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4: Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

Excel

Excel Orientation: The mentor needs to tell the importance of MS office or equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel — Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

Task 2 : Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP

Task 3: Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

Powerpoint

Task 1: Students will be working on basic power point utilities and tools which help them create basic powerpoint presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

Task 2: Interactive presentations - Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

REFERENCE BOOKS:

1. Comdex Information Technology course tool kit Vikas Gupta, *WILEY Dreamtech*
2. The Complete Computer upgrade and repair book, 3rd edition Cheryl A Schmidt, *WILEY Dreamtech*
3. Introduction to Information Technology, ITL Education Solutions limited, *Pearson Education.*
4. PC Hardware - A Handbook – Kate J. Chase *PHI* (Microsoft)
5. LaTeX Companion – Leslie Lamport, *PHI/Pearson.*
6. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. — *CISCO Press, Pearson Education*.
7. IT Essentials PC Hardware and Software Labs and Study Guide Third Edition by Patrick Regan — CISCO Press, *Pearson Education*.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 103L: OBJECT ORIENTED PROGRAMMING LANGUAGE LAB

B.Tech. I Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic Knowledge of computer

Course Objective:

1. To reinforce object-oriented programming concepts through practical implementation.
2. To provide hands-on experience with C++ and Java programming.
3. To develop problem-solving skills using object-oriented techniques.
4. To enable students to design and implement software solutions using OOP principles.
5. To familiarize students with debugging and testing in C++ and Java environments.

Learning Course Outcome:

After successful completion of the course, students will able to:

CO1: Apply object-oriented programming concepts to solve problems in C++ and Java.

CO2: Design and implement classes and objects using C++ and Java.

CO3: Utilize inheritance and polymorphism to create reusable and maintainable code.

CO4: Implement encapsulation and abstraction to enhance software robustness.

CO5: Handle exceptions and perform file I/O operations in C++ and Java.

CO6: Use advanced features such as templates, generics, and multithreading in software development.

CO7: Develop and test object-oriented applications through a mini-project.

Course Contents

List of Experiments	60 Hours

Experiment 1: Introduction to Development Environments

- Setting up development environments for C++ and Java.
- Writing and running simple programs in C++ and Java.

Experiment 2: Classes and Objects

- Define classes and create objects in C++ and Java.
- Implement constructors, destructors (in C++), and methods.
- Demonstrate the use of access specifiers.

Experiment 3: Inheritance

- Implement single, multiple (C++), and multilevel inheritance.
- Use constructor and destructor calls in inheritance.
- Demonstrate method overriding in Java and virtual functions in C++.

Experiment 4: Polymorphism

- Implement function overloading and operator overloading in C++.
- Implement method overloading in Java.
- Create and use abstract classes and interfaces in Java.
- Demonstrate runtime polymorphism in C++ and Java.

Experiment 5: Encapsulation and Abstraction

- Implement encapsulation using access specifiers in C++ and Java.
- Design and use abstract classes in C++ and Java.
- Implement packages in Java and namespaces in C++.

Experiment 6: Exception Handling

- Implement try, catch, and finally blocks in Java.
- Use throw and throws in Java.
- Implement exception handling in C++ using try, catch, and throw.

Experiment 7: File Handling

- Perform file read/write operations in C++ using fstream.
- Perform file read/write operations in Java using FileReader, FileWriter, BufferedReader, and BufferedWriter.

Experiment 8: Templates and Generics

- Implement function templates and class templates in C++.
- Use generics in Java to create type-safe collections.

Experiment 9: Standard Template Library (STL) in C++

- Use various components of STL such as vectors, lists, and maps.
- Perform operations using iterators and algorithms provided by STL.

Experiment 10: Collections Framework in Java

- Use various collection classes such as ArrayList, LinkedList, HashSet, and HashMap.
- Perform operations using iterators and methods provided by the collections framework.

Experiment 11: Multithreading

- Create and manage threads in C++ using the standard thread library.
- Implement multithreading in Java using the Thread class and Runnable interface.

Experiment 12: Mini Project

- Develop a mini-project using C++ or Java to apply object-oriented programming concepts learned throughout the course. This could be a small application such as a student management system, a simple game, or any other real-world application.

REFERENCE BOOKS:

1. "The Complete Reference C++" by Herbert Schildt ISBN-13: 978-0071634809
2. "Effective C++: 55 Specific Ways to Improve Your Programs and Designs" by Scott Meyers ISBN-13: 978-0321334879
3. "Thinking in Java" by Bruce Eckel ISBN-13: 978-0131872486
4. "Effective Java" by Joshua Bloch ISBN-13: 978-0134685991
5. "Java: The Complete Reference" by Herbert Schildt ISBN-13: 978-1260440232

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1: Apply object-oriented programming concepts to solve problems in C++ and Java	3	3	2	2	3	-	-	2	2	2
CO2: Design and implement classes and objects using C++ and Java	3	3	2	2	3	-	-	3	3	3
CO3: Utilize inheritance and polymorphism to create reusable and maintainable code	3	3	2	2	3	-	-	3	3	3
CO4: Implement encapsulation and abstraction to enhance software robustness	3	3	2	2	3	-	-	3	3	3
CO5: Handle exceptions and perform file I/O operations in C++ and Java	3	3	2	2	3	-	-	3	3	3
CO6: Use advanced features such as templates, generics, and multithreading in software development	3	3	2	2	3	-	-	3	3	3
CO7: Develop and test object-oriented applications through a mini-project	3	3	2	2	3	-	-	3	3	3

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 105L: ENGLISH LANGUAGE AND COMMUNICATION SKILLS LABORATORY

B.Tech. I Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: 10+2 Level Education

Course Objective:

1. To facilitate computer-assisted multi-media instruction enabling individualized and independent language learning
2. To sensitize the students to the nuances of English speech sounds, word accent, intonation and rhythm
3. To bring about a consistent accent and intelligibility in students' pronunciation of English by providing an opportunity for practice in speaking
4. To improve the fluency of students in spoken English and neutralize the impact of dialects.
5. To train students to use language appropriately for public speaking, group discussions and interviews.

Learning Course Outcome:

After successful completion of the course, students will able to:

1. Understand the nuances of English language through audio- visual experience and group activities
2. Neutralise their accent for intelligibility
3. Speak with clarity and confidence which in turn enhances their employability skills

Course Contents

UNIT-I	Computer Assisted Language Learning (CALL) Lab	07 Hours

The following course content is prescribed for the **English Language and Communication Skills Lab**. **Exercise – I**

CALL Lab:

Understand: Listening Skill- Its importance – Purpose- Process- Types- Barriers- Effective Listening. *Practice*: Introduction to Phonetics – Speech Sounds – Vowels and Consonants – Minimal Pairs- Consonant Clusters- Past Tense Marker and Plural Marker-

Testing Exercises

ICS Lab:

Understand: Spoken vs. Written language- Formal and Informal English.

Practice: Ice-Breaking Activity and JAM Session- Situational Dialogues – Greetings – Taking Leave – Introducing Oneself and Others.

Exercise – II CALL Lab:

Understand: Structure of Syllables – Word Stress– Weak Forms and Strong Forms – Stress pattern in sentences — Intonation.

Practice: Basic Rules of Word Accent - Stress Shift - Weak Forms and Strong Forms- Stress pattern in sentences — Intonation - *Testing Exercises*

ICS Lab:

Understand: Features of Good Conversation – Strategies for Effective Communication.

Practice: Situational Dialogues – Role Play- Expressions in Various Situations –Making Requests and Seeking Permissions - Telephone Etiquette.

Exercise - III CALL Lab:

Understand: Errors in Pronunciation-Neutralising Mother Tongue Interference (MTI).

Practice: Common Indian Variants in Pronunciation – Differences between British and American Pronunciation -*Testing Exercises*

ICS Lab:

Understand: Descriptions- Narrations- Giving Directions and Guidelines – Blog Writing

Practice: Giving Instructions – Seeking Clarifications – Asking for and Giving Directions – Thanking and Responding – Agreeing and Disagreeing – Seeking and Giving Advice – Making Suggestions.

Exercise – IV CALL Lab:

Understand: Listening for General Details.

Practice: Listening Comprehension Tests - *Testing Exercises*

ICS Lab:

Understand: Public Speaking — Exposure to Structured Talks - Non-verbal Communication- Presentation Skills.

Practice: Making a Short Speech – Extempore- Making a Presentation.

Exercise – V CALL Lab:

Understand: Listening for Specific Details.

Practice: Listening Comprehension Tests -*Testing Exercises*

ICS Lab:

Understand: Group Discussion

Practice: Group Discussion

Minimum Requirement of infrastructural facilities for ELCS Lab:

1. **Computer Assisted Language Learning (CALL) Lab:**

The Computer Assisted Language Learning Lab has to accommodate 40 students with 40 systems, with one Master Console, LAN facility and English language learning software for self- study by students.

System Requirement (Hardware component):

Computer network with LAN facility (minimum 40 systems with multimedia) with the following specifications:

i) Computers with Suitable Configuration
ii) High Fidelity Headphones

2. **Interactive Communication Skills (ICS) Lab :**

The Interactive Communication Skills Lab: A Spacious room with movable chairs and audio- visual aids with a Public Address System, a T. V. or LCD, a digital stereo –audio & video system and camcorder etc.

Source of Material (Master Copy):

- *Exercises in Spoken English. Part 1,2,3*. CIEFL and Oxford University Press

Note: Teachers are requested to make use of the master copy and get it tailor-made to suit the contents of the syllabus.

Suggested Software:

- Cambridge Advanced Learners’ English Dictionary with CD.
- Grammar Made Easy by Darling Kindersley.
- Punctuation Made Easy by Darling Kindersley.
- Oxford Advanced Learner’s Compass, 10th Edition.
- English in Mind (Series 1-4), Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge.
- English Pronunciation in Use (Elementary, Intermediate, Advanced) Cambridge University Press.
- English Vocabulary in Use (Elementary, Intermediate, Advanced) Cambridge University Press.
- TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS).
- Digital All Orell Digital Language Lab (Licensed Version)

REFERENCE BOOKS:

1. (2022). *English Language Communication Skills – Lab Manual cum Workbook.* Cengage Learning India Pvt. Ltd.
2. Shobha, KN & Rayen, J. Lourdes. (2019). *Communicative English – A workbook.* Cambridge University Press
3. Kumar, Sanjay & Lata, Pushp. (2019). *Communication Skills: A Workbook.* Oxford University Press
4. Board of Editors. (2016). *ELCS Lab Manual: A Workbook for CALL and ICS Lab Activities.* Orient Black Swan Pvt. Ltd.
5. Mishra, Veerendra et al. (2020). *English Language Skills: A Practical Approach.* Cambridge University Press.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 201: APPLIED STATISTICS
B.Tech. I Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic understanding of calculus, linear algebra, and probability theory.

Course Objective:

1. To understand the fundamental concepts and principles of applied statistics.
2. To learn how to collect, organize, and summarize data effectively.
3. To develop proficiency in probability theory and distributions.
4. To master techniques of inferential statistics for hypothesis testing and estimation.
5. To apply regression analysis and other statistical methods for predictive modeling.
6. To gain experience in using statistical software for data analysis.
7. To interpret and communicate statistical findings effectively.
8. To enhance problem-solving skills through real-world statistical applications.
9. To prepare students for advanced study and professional work in statistics and related fields.
10. To introduce advanced statistical techniques such as non-parametric tests, cluster analysis, and factor analysis.

Learning Course Outcome:

After successful completion of the course, students will able to:

1. Explain and apply the basic concepts of descriptive and inferential statistics.
2. Collect, organize, and summarize data using appropriate statistical methods.
3. Use probability theory to model and analyze random phenomena.
4. Conduct hypothesis testing and construct confidence intervals.
5. Perform and interpret regression analysis and other predictive modeling techniques.
6. Utilize statistical software such as R, SPSS, or Python for data analysis.
7. Analyze real-world data sets and derive meaningful insights.
8. Communicate statistical results effectively through written reports and presentations.
9. Apply statistical techniques to solve practical problems in various domains.
10. Apply advanced statistical techniques such as non-parametric tests, cluster analysis, and factor analysis to real-world data sets.

Course Contents

UNIT-I	Introduction to Applied Statistics	08 Hours

Overview of Statistics
- 1.1.1 Definition of Statistics
  - What is Statistics?
  - Branches of Statistics: Descriptive vs. Inferential Statistics
- 1.1.2 Role of Statistics in Research
  - Importance of Statistics in Scientific Research
  - Examples of Statistical Applications in Various Fields
- 1.1.3 Statistical Thinking
  - Understanding Variability
  - Data-Driven Decision Making
- 1.1.4 Key Statistical Concepts
  - Population and Sample
  - Parameter and Statistic
  - Variables: Qualitative vs. Quantitative
- 1.1.5 The Statistical Process
  - Steps in the Statistical Process: Collection, Analysis, Interpretation, Presentation

UNIT-II	**Inferential Statistics**	**10 Hours**

4.1 Sampling and Sampling Distributions
- 4.1.1 Introduction to Sampling
  - Definition and Importance of Sampling
  - Difference Between Population and Sample
  - Types of Sampling Methods: Random, Stratified, Systematic, Cluster
- 4.1.2 Sampling Distributions
  - Definition and Concept of a Sampling Distribution
  - Central Limit Theorem (CLT)
  - Properties of Sampling Distributions of Sample Mean and Sample Proportion
- 4.1.3 Sampling Techniques
  - Practical Approaches to Implementing Sampling Methods
  - Examples and Applications of Different Sampling Techniques
- 4.1.4 Examples and Applications
  - Real-world Scenarios Involving Sampling and Sampling Distributions

4.2 Point Estimation and Interval Estimation
- 4.2.1 Point Estimation
  - Definition and Purpose of Point Estimation
  - Properties of Point Estimators: Unbiasedness, Consistency, Efficiency
  - Common Point Estimators: Mean, Variance, Proportion
- 4.2.2 Interval Estimation
  - Concept and Importance of Interval Estimation
  - Confidence Intervals: Interpretation and Calculation
  - Factors Affecting the Width of Confidence Intervals
- 4.2.3 Methods for Constructing Interval Estimates
  - Construction of Confidence Intervals for Means, Proportions, and Variances
  - Use of Z-scores and t-scores in Interval Estimation
- 4.2.4 Examples and Applications

Practical Examples of Point and Interval Estimations 4.3 Hypothesis Testing 4.3.1 Null and Alternative Hypotheses 4.3.1.1 Formulating Hypotheses Definition of Null and Alternative Hypotheses Setting Up Hypotheses for Different Types of Tests One-Tailed vs. Two-Tailed Tests 4.3.1.2 Examples of Hypothesis Formulation Real-world Scenarios and Formulating Appropriate Hypotheses 4.3.2 Type I and Type II Errors 4.3.2.1 Understanding Errors in Hypothesis Testing Definition and Examples of Type I Error (False Positive) Definition and Examples of Type II Error (False Negative) 4.3.2.2 Balancing Type I and Type II Errors Relationship Between the Errors Significance Level (Alpha) and Power of the Test (1 - Beta) 4.3.2.3 Examples and Applications Case Studies Demonstrating Type I and Type II Errors 4.3.3 p-values and Significance Levels 4.3.3.1 Introduction to p-values Definition and Interpretation of p-values Calculating p-values for Different Statistical Tests 4.3.3.2 Significance Levels and Decision Making Setting Significance Levels (Alpha) Making Decisions Based on p-values and Significance Levels 4.3.3.3 Examples and Applications Real-world Examples of Hypothesis Testing Using p-values and Significance Levels 4.4 Confidence Intervals 4.4.1 Confidence Interval for Means 4.4.1.1 Constructing Confidence Intervals for Means Using Z-scores for Known Population Standard Deviation Using t-scores for Unknown Population Standard Deviation 4.4.1.2 Examples and Applications Practical Examples Involving Confidence Intervals for Means 4.4.2 Confidence Interval for Proportions 4.4.2.1 Constructing Confidence Intervals for Proportions Formula and Calculation Steps Considerations for Large and Small Sample Sizes 4.4.2.2 Examples and Applications Practical Examples Involving Confidence Intervals for Proportions		
UNIT-III	**Regression Analysis**	**10 Hours**

5.1 Simple Linear Regression
- 5.1.1 Model Assumptions
  - Linearity: The relationship between the independent and dependent variable is linear.
  - Independence: Observations are independent of each other.
  - Homoscedasticity: Constant variance of the errors (residuals).
  - Normality: The residuals of the model are normally distributed.
- 5.1.2 Estimation of Parameters
  - Least Squares Method:
    - Concept of minimizing the sum of squared residuals.
    - Calculation of slope (β1) and intercept (β0) parameters.
  - Interpretation of Parameters:
    - Understanding the meaning of the slope and intercept in the context of the model.
  - Confidence Intervals for Parameters:
    - Constructing and interpreting confidence intervals for β0 and β1.
- 5.1.3 Goodness of Fit: R-squared
  - Definition and Calculation:
    - Understanding the proportion of variance explained by the model.
  - Adjusted R-squared:
    - Adjusting R-squared for the number of predictors in the model.
  - Interpreting R-squared:
    - Practical significance and limitations of R-squared as a measure of fit.

5.2 Multiple Linear Regression
- 5.2.1 Model Building
  - Incorporating Multiple Predictors:
    - Extending simple linear regression to include multiple predictors.
  - Model Specification:
    - Selecting relevant variables and model form.
  - Assumptions in Multiple Regression:
    - Checking for multicollinearity and its impact on the model.
- 5.2.2 Interpretation of Coefficients
  - Understanding Coefficients:
    - Partial regression coefficients and their interpretation.
  - Statistical Significance:
    - T-tests for individual coefficients and p-values.
  - Confidence Intervals:
    - Constructing and interpreting confidence intervals for regression coefficients.
- 5.2.3 Diagnostics and Residual Analysis
  - Residual Plots:
    - Checking for homoscedasticity and non-linearity.
  - Influence and Leverage:
    - Identifying influential observations using Cook's distance and leverage statistics.
  - Multicollinearity Diagnostics:
    - Variance Inflation Factor (VIF) and tolerance.

5.3 Advanced Regression Techniques
- 5.3.1 Logistic Regression
  - Concept and Application:

Understanding binary outcomes and the logistic function.
Model Estimation:
Maximum likelihood estimation of parameters.
Odds Ratios and Interpretation:
Interpreting the coefficients as odds ratios.
Model Fit and Validation:
Hosmer-Lemeshow test, ROC curves, and AUC.
5.3.2 Polynomial Regression
Extending Linear Models:
Introduction to polynomial terms in regression.
Model Specification:
Choosing the degree of the polynomial and interpreting coefficients.
Curve Fitting:
Comparing polynomial regression to linear regression.
Overfitting and Model Complexity:
Understanding the trade-off between bias and variance.

UNIT-IV	**Advanced Statistical Techniques**	**07 Hours**

6.1 Non-parametric Tests
Non-parametric tests are statistical methods used when data do not fit the assumptions of parametric tests, such as normal distribution or homogeneity of variance. They are useful for analyzing ordinal data or data that are not normally distributed.
6.1.1 Chi-Square Test
Introduction and Application:
Overview of the Chi-Square test for independence and goodness-of-fit.
Chi-Square Test for Independence:
Testing the independence between two categorical variables.
Constructing and interpreting contingency tables.
Calculating expected frequencies.
Interpreting Chi-Square statistics and p-values.
Chi-Square Goodness-of-Fit Test:
Testing if a sample distribution fits an expected distribution.
Calculating observed and expected frequencies.
Degrees of freedom and significance testing.
6.1.2 Mann-Whitney U Test
Introduction and Application:
Non-parametric alternative to the independent samples t-test.
Suitable for comparing differences between two independent groups.
Procedure:
Ranking all the observations from both groups together.
Calculating the U statistic and its distribution.
Interpreting the test results and p-values.
Assumptions and Limitations:
Discussing the assumptions of the Mann-Whitney U test.
Comparison with the t-test and scenarios where it is preferred.

6.1.3 Kruskal-Wallis Test

Introduction and Application:

Non-parametric alternative to one-way ANOVA.

Suitable for comparing differences between more than two independent groups.

Procedure:

Ranking all observations across all groups.

Calculating the H statistic and its distribution.

Interpreting the test results and p-values.

Assumptions and Limitations:

Discussing the assumptions of the Kruskal-Wallis test.

Post-hoc analysis for pairwise comparisons.

6.2 Multivariate Analysis

Multivariate analysis involves examining multiple variables simultaneously to understand relationships and patterns in the data.

6.2.1 Cluster Analysis

Introduction and Types:

Overview of clustering techniques: Hierarchical and K-means clustering.

Hierarchical Clustering:

Concepts of agglomerative and divisive clustering.

Dendrogram construction and interpretation.

Linkage criteria: single, complete, average, and Ward’s method.

K-means Clustering:

Algorithm steps and iteration process.

Selecting the number of clusters (k).

Interpretation of clustering results.

Applications and Interpretation:

Practical applications in market segmentation, image analysis, etc.

Validating and interpreting clusters.

6.2.2 Factor Analysis

Introduction and Objectives:

Identifying underlying factors that explain the data structure.

Differentiating between exploratory and confirmatory factor analysis.

Exploratory Factor Analysis (EFA):

Steps involved: extraction, rotation, and interpretation.

Methods of extraction: Principal Axis Factoring, Maximum Likelihood.

Rotation methods: Varimax, Promax.

Confirmatory Factor Analysis (CFA):

Model specification and testing.

Goodness-of-fit indices.

Applications and Interpretation:

Applications in psychology, social sciences, and marketing research.

Interpretation of factor loadings and communalities.

6.2.3 Principal Component Analysis (PCA)

Introduction and Objectives:

Reducing dimensionality while preserving variance.

Difference between PCA and Factor Analysis.

Procedure:
- Standardizing the data.
- Calculating the covariance matrix and eigenvalues/eigenvectors.
- Determining the number of principal components.

Interpreting PCA Results:
- Scree plot and explained variance.
- Loading plots and score plots.

Applications and Interpretation:
- Applications in data compression, visualization, and feature extraction.
- Practical examples and interpretation of components.

UNIT-V	Statistical Software Applications	08 Hours

7.1 Introduction to Statistical Software

Overview of Statistical Software:
- Importance and benefits of using statistical software in data analysis.
- Comparison of popular statistical software: R, SPSS, and Python.

Choosing the Right Tool:
- Criteria for selecting appropriate statistical software based on the type of analysis and user expertise.

Installation and Setup:
- Step-by-step guide for installing R, SPSS, and Python.
- Introduction to the user interface of each software.

7.2 Data Analysis Using R

R is a powerful tool for statistical analysis and data visualization. It is widely used in academia and industry due to its versatility and extensive library support.

7.2.1 Data Manipulation

Data Importing:
- Reading data from various sources: CSV, Excel, databases, and web.

Data Cleaning:
- Handling missing values and duplicates.
- Data transformation: filtering, sorting, and summarizing.

Data Structures:
- Overview of data frames, lists, and matrices.
- Applying functions to manipulate data structures.

7.2.2 Statistical Analysis

Descriptive Statistics:
- Calculating measures of central tendency and dispersion.
- Generating summary statistics for data exploration.

Inferential Statistics:
- Performing hypothesis tests: t-tests, chi-square tests.
- Constructing confidence intervals.

Regression Analysis:
- Implementing linear and logistic regression models.
- Interpreting model coefficients and diagnostic plots.

7.2.3 Visualization

Basic Plots:

Creating histograms, bar plots, and box plots.

Advanced Visualization:

Using ggplot2 for complex and customizable plots.

Plotting time series data and multi-panel plots.

Interactive Visualization:

Introduction to Shiny for building interactive web applications.

7.3 Data Analysis Using SPSS

SPSS (Statistical Package for the Social Sciences) is user-friendly software widely used for statistical analysis in social sciences and other fields.

7.3.1 Data Entry and Management

Data Entry:

Manual data entry and importing data from external files.

Defining variables and variable types.

Data Management:

Labeling data and creating value labels.

Data transformation: recoding and computing new variables.

7.3.2 Descriptive Statistics

Generating Descriptive Statistics:

Calculating mean, median, mode, and standard deviation.

Creating frequency distributions and cross-tabulations.

Graphical Representation:

Producing histograms, pie charts, and scatter plots.

Customizing graphs and charts for publication-quality output.

7.3.3 Inferential Statistics

Hypothesis Testing:

Performing t-tests, ANOVA, and chi-square tests.

Interpreting test results and p-values.

Regression Analysis:

Conducting linear and logistic regression.

Evaluating model fit and assumptions.

Non-parametric Tests:

Implementing Mann-Whitney U test and Kruskal-Wallis test.

Analyzing data with non-parametric methods.

7.4 Data Analysis Using Python

Python is a versatile programming language with powerful libraries for data analysis and visualization. It is popular for its readability and extensive community support.

7.4.1 Libraries: NumPy, Pandas, Matplotlib, SciPy

Introduction to Libraries:

Overview of NumPy for numerical computations.

Pandas for data manipulation and analysis.

Matplotlib for data visualization.

SciPy for scientific and technical computing.

7.4.2 Data Cleaning and Preparation

Data Importing:

Reading data from CSV, Excel, SQL databases, and web APIs.

Data Cleaning:

Handling missing data and outliers.
Data transformation: filtering, merging, and grouping.
Data Preparation:
Creating new features and engineering existing ones.
Normalizing and standardizing data.
7.4.3 Statistical Modeling and Analysis
Descriptive Statistics:
Calculating summary statistics and generating descriptive reports.
Inferential Statistics:
Conducting hypothesis tests and calculating confidence intervals.
Regression Analysis:
Building and evaluating linear and logistic regression models.
Using statsmodels and scikit-learn for statistical modeling.
Advanced Analysis:
Implementing time series analysis and forecasting.
Performing cluster analysis and principal component analysis (PCA).

UNIT-VI	**Real-World Applications and Case Studies**	**08 Hours**

8.1 Business Analytics
Business analytics involves applying statistical techniques to analyze and interpret business data to make informed decisions. This section focuses on practical applications in various business contexts.
8.1.1 Descriptive Analytics in Business
Sales and Revenue Analysis:
Using descriptive statistics to analyze sales data.
Identifying trends and patterns in revenue streams.
Customer Segmentation:
Segmenting customers based on purchasing behavior.
Using clustering techniques for market segmentation.
8.1.2 Predictive Analytics in Business
Sales Forecasting:
Implementing regression models to predict future sales.
Evaluating model accuracy and making adjustments.
Churn Prediction:
Applying logistic regression to predict customer churn.
Interpreting model results to develop retention strategies.
8.1.3 Case Study: Financial Risk Management
Risk Assessment:
Analyzing financial data to assess risk.
Using statistical models to predict and mitigate financial risks.
Portfolio Optimization:
Applying statistical methods to optimize investment portfolios.
Balancing risk and return through diversification strategies.
8.2 Engineering Statistics
Engineering statistics involves the application of statistical methods to solve engineering problems. This section covers key applications in quality control, reliability analysis, and process optimization.

8.2.1 Quality Control and Six Sigma
Control Charts:
Constructing and interpreting control charts for process monitoring.
Identifying special cause variations and taking corrective actions.
Process Capability Analysis:
Assessing process capability using Cp and Cpk indices.
Improving process performance through statistical analysis.
8.2.2 Reliability Engineering
Reliability Modeling:
Estimating product reliability using life data analysis.
Applying Weibull analysis for reliability assessment.
Failure Mode and Effect Analysis (FMEA):
Identifying and prioritizing potential failure modes.
Using statistical techniques to mitigate risks and improve reliability.
8.2.3 Case Study: Manufacturing Process Improvement
Lean Manufacturing:
Analyzing production data to identify waste and inefficiencies.
Implementing statistical methods to streamline processes.
Process Optimization:
Using Design of Experiments (DoE) for process optimization.
Enhancing product quality and reducing production costs.
8.3 Social Science Research
Social science research utilizes statistical methods to study human behavior, social phenomena, and societal trends. This section explores applications in survey analysis, experimental design, and data interpretation.
8.3.1 Survey Design and Analysis
Questionnaire Design:
Designing effective survey instruments.
Ensuring validity and reliability of survey questions.
Sampling Techniques:
Implementing random and stratified sampling methods.
Calculating sample size and ensuring representativeness.
8.3.2 Experimental Design
Design of Experiments:
Planning and conducting experiments in social sciences.
Analyzing experimental data using ANOVA and regression techniques.
Quasi-Experimental Designs:
Applying quasi-experimental designs for causal inference.
Addressing confounding variables and bias in social research.
8.3.3 Case Study: Public Opinion Research
Opinion Polls:
Conducting and analyzing opinion polls.
Interpreting poll results and assessing public sentiment.
Behavioral Studies:
Analyzing behavioral data to understand social trends.
Using statistical models to predict social behaviors.

8.4 Health Sciences

Health sciences apply statistical methods to study medical data, analyze clinical trials, and improve public health outcomes. This section focuses on biostatistics, epidemiology, and health data analysis.

8.4.1 Biostatistics

Clinical Trials:

Designing and analyzing clinical trials.

Applying statistical methods to evaluate treatment efficacy.

Survival Analysis:

Conducting survival analysis using Kaplan-Meier curves and Cox proportional hazards models.

Interpreting survival data and making clinical recommendations.

8.4.2 Epidemiology

Epidemiological Studies:

Designing cohort, case-control, and cross-sectional studies.

Analyzing epidemiological data to identify risk factors and disease patterns.

Disease Surveillance:

Monitoring and analyzing disease outbreak data.

Applying statistical techniques for disease prevention and control.

8.4.3 Case Study: Public Health Data Analysis

Health Surveys:

Analyzing data from national health surveys.

Identifying health disparities and priority areas for intervention.

Health Policy Evaluation:

Evaluating the impact of health policies using statistical methods.

Making evidence-based recommendations for policy improvements.

REFERENCE BOOKS:

1. "Numerical Methods for Engineers" by Steven C. Chapra and Raymond P. Canale
2. "Applied Numerical Methods with MATLAB" by Steven C. Chapra
3. "Elements of Information Theory" by Thomas M. Cover and Joy A. Thomas
4. "Information Theory, Inference, and Learning Algorithms" by David J.C. MacKay

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Mark

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 202: PROGRAMMING FOR PROBLEM SOLVING
B.Tech. I Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	**TH: 04**	**In Sem. Evaluation: 25 Marks** **Mid Sem. Exam: 25 Marks** **End Sem. Exam: 50 Marks** **Total : 100 Marks**

Course Prerequisites: 10 + 2 Science

Course Objective:

1. To introduce the fundamental concepts of programming using Python.
2. To develop problem-solving skills through programming.
3. To teach the basic constructs of Python for algorithmic thinking.
4. To familiarize students with Python’s data structures and standard libraries.
5. To enable students to write efficient, readable, and modular code.
6. To prepare students for advanced courses in data science, machine learning, and software development.

Learning Course Outcome:
After successful completion of the course, students will able to:

CO1: Understand the basic concepts of programming and problem-solving using Python.
CO2: Apply control structures for decision making and looping in Python programs.
CO3: Develop functions and modules for modular and reusable code.
CO4: Utilize Python's built-in data structures for efficient data manipulation.
CO5: Perform file handling operations for reading and writing data.
CO6: Implement object-oriented programming concepts in Python.
CO7: Use basic libraries for data science to manipulate and visualize data.

Course Contents

UNIT-I	Introduction to Python	08 Hours
Introduction to Programming and Problem Solving, Basics of Python: Installing Python, IDEs, Python Interpreter, Python Syntax, Keywords, Identifiers, and Variables, Basic Input/Output Operations, Data Types and Type Conversion, Comments and Documentation		
UNIT-II	**Control Structures**	**08 Hours**
Decision Making: if, if-else, nested if-else, Looping: for, while, nested loops, Control Statements: break, continue, pass, List Comprehensions, Practical Applications of Control Structures.		
UNIT-III	**Functions and Modules**	**08 Hours**

Defining and Calling Functions, Function Arguments and Return Values, Lambda Functions, Recursion, Modules and Packages, Built-in Functions and Standard Libraries.		
UNIT-IV	**Data Structures**	**08 Hours**
Lists: Operations, Methods, and Applications, Tuples: Operations and Applications, Dictionaries: Operations, Methods, and Applications, Sets: Operations and Methods, String Manipulation and Methods, Arrays and their Applications.		
UNIT-V	**File Handling**	**08 Hours**
Reading from and Writing to Files, File Methods and Attributes, Working with Text and Binary Files, Exception Handling in File Operations, CSV File Handling.		
UNIT-VI	**Advanced Topics**	**08 Hours**
Object-Oriented Programming in Python: Classes and Objects, Inheritance and Polymorphism, Encapsulation and Abstraction, Introduction to Libraries for Data Science: NumPy: Arrays and Matrices, Pandas: DataFrames and Series, Matplotlib: Basic Plotting Introduction to Regular Expressions		

REFERENCE BOOKS:

1. "Python Crash Course" by Eric Matthes ISBN-13: 978-1593276034
2. "Automate the Boring Stuff with Python" by Al Sweigart ISBN-13: 978-1593275990
3. "Learning Python" by Mark Lutz ISBN-13: 978-1449355739
4. "Python for Data Analysis" by Wes McKinney ISBN-13: 978-1491957660

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1: Understand the basic concepts of programming and problem-solving using Python	3	3	2	2	3	-	-	2	2	2
CO2: Apply control structures for decision making and looping in Python programs	3	3	2	2	3	-	-	3	3	3
CO3: Develop functions and modules for modular and reusable code	3	3	2	2	3	-	-	3	3	3
CO4: Utilize Python's built-in data structures for efficient data manipulation	3	3	2	2	3	-	-	3	3	3
CO5: Perform file handling operations for reading and writing data	3	3	2	2	3	-	-	3	3	3
CO6: Implement object-oriented	3	3	2	2	3	-	-	3	3	3

programming concepts in Python										
CO7: Use basic libraries for data science to manipulate and visualize data	**3**	**3**	**2**	**2**	**3**	**-**	**-**	**3**	**3**	**3**

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 203: Data Security in Data Science
B.Tech. I Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic understanding of computer networks and operating systems, Familiarity with data science concepts, tools, and programming languages.

Course Objective:

- To understand the fundamental concepts of data security in the context of data science.
- To learn various cryptographic techniques and their applications in data science.
- To explore methods for securing data during storage, transmission, and processing in data science workflows.
- To examine security protocols and standards relevant to data science.
- To gain practical experience in implementing data security measures in data science projects.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Demonstrate an understanding of data security principles and practices in data science.
- Apply cryptographic techniques to secure data used in data science.
- Implement security measures for data storage, transmission, and processing in data science workflows.
- Evaluate and use security protocols and standards in data science projects.
- Analyze security risks in data science and develop strategies to mitigate them.

Course Contents

UNIT-I	Introduction to Data Security in Data Science	07 Hours
Overview of Data Security: Importance and Challenges in Data Science, Basic Concepts: Confidentiality, Integrity, Availability, Threats and Vulnerabilities in Data Science: Types and Examples, Risk Management and Assessment in Data Science Projects, Legal and Ethical Issues in Data Security for Data Science		
UNIT-II	**Cryptography for Data Science**	**08 Hours**
Introduction to Cryptography: Goals and Applications in Data Science, Symmetric Key Cryptography: Algorithms (AES, DES), Asymmetric Key Cryptography: Algorithms (RSA, ECC), Hash Functions and Digital Signatures, Key Management and Distribution in Data Science		
UNIT-III	**Data Security in Storage for Data Science**	**08 Hours**

Data Encryption at Rest: Techniques and Tools, Secure Data Storage Solutions: Databases, Data Lakes, and Cloud Storage, Access Controls and Authentication Mechanisms, Backup and Recovery Strategies for Data Science Projects, Case Studies: Implementing Data Security in Storage for Data Science		
UNIT-IV	**Data Security in Transmission for Data Science**	**07 Hours**
Data Encryption in Transit: Techniques and Tools, Secure Communication Protocols: SSL/TLS, HTTPS, Network Security Measures: Firewalls, Intrusion Detection Systems, Virtual Private Networks (VPNs) for Data Science, Case Studies: Implementing Data Security in Transmission for Data Science		
UNIT-V	**Security Protocols and Standards for Data Science**	**08 Hours**
Overview of Security Protocols: Importance and Applications in Data Science, Authentication Protocols: Kerberos, OAuth, Secure Data Sharing Protocols: Secure Multiparty Computation, Differential Privacy, Security Standards and Frameworks: ISO/IEC 27001, NIST, GDPR, Case Studies: Implementing Security Protocols and Standards in Data Science Projects		
UNIT-VI	**Advanced Topics and Emerging Trends in Data Security for Data Science**	**08 Hours**
Advanced Encryption Techniques: Quantum Cryptography, Homomorphic Encryption, Data Privacy and Anonymization Techniques: GDPR, CCPA Compliance, Cloud Security for Data Science: Challenges and Solutions, Security in Big Data and IoT: Techniques and Best Practices, Future Trends in Data Security for Data Science		

TEXT BOOKS:

1. "Cryptography and Network Security: Principles and Practice" by William Stallings
2. "Data Security and Privacy: A Practical Guide for Protecting Data" by David Salomon
3. "Security and Privacy in Data Mining and Machine Learning" by Charu C. Aggarwal
4. "Applied Cryptography: Protocols, Algorithms, and Source Code in C" by Bruce Schneier
5. "Database Security: Concepts, Approaches, and Challenges" by Alfred Basta and Melissa Zgola

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcome	PO1	PO2	PO3	PO4	PO6
CO1: Understand data security principles	2	2	-	-	3
CO2: Apply cryptographic techniques	2	-	3	-	-
CO3: Implement security measures	-	2	3	-	-
CO4: Evaluate security protocols and standards	-	2	-	3	-
CO5: Analyze security risks	-	2	-	-	3

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 204: SOFTWARE ENGINEERING
B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Software Flow Basic

Course Objective:

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Learning Course Outcome:
After successful completion of the course, students will able to:

- Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

Course Contents

UNIT-I	Introduction to Software Engineering	07 Hours

The evolving role of software, changing nature of software, software myths. **A Generic view of process**: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI). **Process models**: The waterfall model, Spiral model and Agile methodology

UNIT-II	Software Requirements	08 Hours

Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.
Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT-III	Design Engineering	08 Hours
Design process and design quality, design concepts, the design model. Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.		
UNIT-IV	**Testing Strategies**	**07 Hours**
A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging. Metrics for Process and Products: Software measurement, metrics for software quality.		
UNIT-V	**Risk management**	**08 Hours**
Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM. **Quality Management:** Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.		

TEXT BOOKS:

1. Software Engineering, A practitioner’s Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.

REFERENCE BOOKS:

1. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students’ reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 202L: PROGRAMMING FOR PROBLEM SOLVING LABORATORY

B.Tech. I Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Familiarity with basic programming concepts and syntax, preferably in another language, and an understanding of fundamental computer science principles.

Course Objective:

To impart the basic concepts of data structures and algorithms. To understand concepts about searching and sorting techniques. To Understand basic concepts about stacks, queues, lists, trees and graphs To understand the algorithms and develop the step by step solutions of problemswith the help of data

Learning Course Outcome:

After successful completion of the course, students will able to:

The experiments will make the student gain skills on:

CO1: Understand Basic Programming Constructs
CO2: Implement Data Structures and Algorithms
CO3: Develop Object-Oriented Programming Skills
CO4: Use Python Libraries for Data Handling and Analysis
CO5: Apply File Handling and Exception Handling Techniques
CO6: Implement and Analyze Sorting and Searching Algorithms
CO7: Develop Skills in Problem Solving and Debugging
CO8: Gain Exposure to Advanced Topics

Course Contents

Experiment 1: Introduction to Python

- Write a Python program to print "Hello, World!".
- Write a Python program to take user input and print it.
- Write a Python program to swap two numbers.

Experiment 2: Control Structures

- Write a Python program to check if a number is even or odd.
- Write a Python program to find the factorial of a number using loops.
- Write a Python program to print Fibonacci series up to n terms.

Experiment 3: Functions and Modules

- Write a Python program to find the largest number among three numbers using a function.
- Write a Python program to check if a number is prime or not using a function.
- Write a Python program to generate a random number between a given range using a module.

Experiment 4: Lists and Strings

- Write a Python program to find the sum and average of elements in a list.
- Write a Python program to count the number of vowels in a string.
- Write a Python program to reverse a string using slicing.

Experiment 5: Dictionaries and Sets

- Write a Python program to demonstrate dictionary operations (add, delete, update).
- Write a Python program to find the intersection and union of two sets.
- Write a Python program to count the frequency of elements in a list using a dictionary.

Experiment 6: File Handling

- Write a Python program to read and display the contents of a file.
- Write a Python program to write user input to a file.
- Write a Python program to count the number of words in a file.

Experiment 7: Object-Oriented Programming (OOP)

- Write a Python program to demonstrate class and object creation.
- Write a Python program to demonstrate inheritance (base and derived classes).
- Write a Python program to demonstrate method overriding.

Experiment 8: Exception Handling

- Write a Python program to handle ZeroDivisionError and IndexError exceptions.

- Write a Python program to demonstrate the use of try-except-else-finally block.
- Write a Python program to create a custom exception and raise it.

Experiment 9: Recursion

- Write a Python program to find the factorial of a number using recursion.
- Write a Python program to calculate the nth Fibonacci number using recursion.
- Write a Python program to implement binary search using recursion.

Experiment 10: Sorting and Searching Algorithms

- Write a Python program to implement bubble sort and analyze its time complexity.
- Write a Python program to implement binary search on a sorted list.
- Write a Python program to implement linear search and analyze its time complexity.

Experiment 11: Data Structures (Optional)

- Write a Python program to implement a stack using lists.
- Write a Python program to implement a queue using lists.
- Write a Python program to implement a binary search tree (BST).

Experiment 12: Introduction to NumPy and Pandas (Data Handling)

- Write a Python program to create a NumPy array and perform basic operations (addition, subtraction).
- Write a Python program to create a Pandas DataFrame and perform basic operations (selection, filtering).

REFERENCE BOOKS:

1. "Python Programming: An Introduction to Computer Science" by John Zelle
2. Official documentation of Python (python.org)
3. Official documentation of NumPy (numpy.org)
4. Official documentation of Pandas (pandas.pydata.org)

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
CO1: Understand Basic Programming Constructs	3	3	-	-	-	3	-	3
CO2: Implement Data Structures and Algorithms	3	3	-	-	-	3	2	3
CO3: Develop Object-Oriented Programming Skills	3	3	2	-	-	2	3	3
CO4: Use Python Libraries for Data Handling and Analysis	3	3	2	2	-	3	-	3
CO5: Apply File Handling and Exception Handling Techniques	3	3	-	2	-	2	-	3
CO6: Implement and Analyze Sorting and Searching Algorithms	3	3	-	-	-	2	2	3
CO7: Develop Skills in Problem Solving and Debugging	3	3	-	-	-	3	2	3
CO8: Gain Exposure to Advanced Topics	3	2	-	-	-	2	2	3

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 201L: APPLIED STATISTICS LAB
B.Tech. I Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic knowledge of Python programming

Course Objective:

1. Introduce students to essential Python packages used in data analytics.
2. Provide hands-on experience in using these packages for data manipulation, analysis, and visualization.
3. Develop the ability to implement data analytics workflows using Python.
4. Enhance problem-solving skills using Python for real-world data analytics problems.
5. Understand the integration of Python packages for comprehensive data science solutions.

Learning Course Outcome:
After successful completion of the course, students will able to:

CO1: Understand and set up the Python environment for data analytics. Know the applications of various devices.

CO2: Demonstrate proficiency in using NumPy and Pandas for data manipulation and preprocessing.

CO3: Create and customize data visualizations using Matplotlib and Seaborn.

CO4: Perform statistical analysis and hypothesis testing using SciPy and Statsmodels.

CO5: Implement machine learning algorithms using Scikit-Learn for both supervised and unsupervised learning tasks.

CO6: Integrate multiple Python packages to develop comprehensive data analytics workflows.

CO7: Explore advanced Python packages for deep learning and apply best practices in data analytics projects.

List of Experiments

Lab Exercises:
Unit 1: Introduction to Applied Statistics
Lab 1: Understanding Types of Data and Levels of Measurement

- Objective: Learn to classify different types of data (nominal, ordinal, interval, ratio) and their levels of measurement.
- Tools: Python (Pandas), R
- Activities: Identify types of data and levels of measurement in a given dataset.

Unit 2: Descriptive Statistics
Lab 2: Data Collection Methods

- Objective: Explore various methods of data collection.
- Tools: Python, R
- Activities: Simulate data collection using surveys, experiments, and observational studies.

Lab 3: Data Organization and Presentation

- Objective: Organize and present data using frequency distributions and graphical representations.
- Tools: Python (Pandas, Matplotlib), R

- Activities: Create frequency distributions, histograms, pie charts, and box plots.

Lab 4: Measures of Central Tendency

- Objective: Calculate mean, median, and mode.
- Tools: Python (Pandas), R
- Activities: Compute and interpret mean, median, and mode for a dataset.

Lab 5: Measures of Dispersion

- Objective: Calculate range, variance, and standard deviation.
- Tools: Python (Pandas), R
- Activities: Compute and interpret range, variance, and standard deviation for a dataset.

Lab 6: Measures of Shape and Position

- Objective: Analyze skewness, kurtosis, percentiles, and quartiles.
- Tools: Python (SciPy, Pandas), R
- Activities: Calculate and interpret skewness, kurtosis, percentiles, and quartiles.

Unit 3: Probability Theory

Lab 7: Basics of Probability

- Objective: Understand basic probability concepts.
- Tools: Python, R
- Activities: Solve probability problems using Python or R.

Lab 8: Probability Distributions

- Objective: Work with discrete and continuous probability distributions.
- Tools: Python (NumPy, SciPy), R
- Activities: Generate and analyze binomial, Poisson, normal, and exponential distributions.

Lab 9: Conditional Probability and Independence

- Objective: Explore conditional probability and independence.
- Tools: Python, R
- Activities: Calculate conditional probabilities and test for independence in datasets.

Unit 4: Inferential Statistics

Lab 10: Sampling and Sampling Distributions

- Objective: Understand sampling techniques and sampling distributions.
- Tools: Python (Pandas, NumPy), R
- Activities: Perform random sampling and create sampling distributions.

Lab 11: Point Estimation and Interval Estimation

- Objective: Perform point and interval estimation.
- Tools: Python (SciPy), R
- Activities: Estimate population parameters using point and interval estimation.

Lab 12: Hypothesis Testing

- Objective: Conduct hypothesis tests.
- Tools: Python (SciPy, Statsmodels), R
- Activities: Perform and interpret t-tests, chi-square tests, and ANOVA.

Lab 13: Confidence Intervals

- Objective: Construct and interpret confidence intervals.
- Tools: Python (SciPy, Statsmodels), R
- Activities: Calculate confidence intervals for means and proportions.

Unit 5: Regression Analysis

Lab 14: Simple Linear Regression

- Objective: Perform simple linear regression analysis.
- Tools: Python (Statsmodels, Scikit-learn), R
- Activities: Fit a simple linear regression model and interpret the results.

Lab 15: Multiple Linear Regression

- Objective: Conduct multiple linear regression analysis.
- Tools: Python (Statsmodels, Scikit-learn), R
- Activities: Build and evaluate a multiple linear regression model.

Lab 16: Advanced Regression Techniques

- Objective: Explore logistic regression and polynomial regression.
- Tools: Python (Scikit-learn), R
- Activities: Fit and interpret logistic and polynomial regression models.

Unit 6: Advanced Statistical Techniques

Lab 17: Non-parametric Tests

- Objective: Conduct non-parametric statistical tests.
- Tools: Python (SciPy), R
- Activities: Perform chi-square, Mann-Whitney U, and Kruskal-Wallis tests.

Lab 18: Cluster Analysis

- Objective: Perform cluster analysis.
- Tools: Python (Scikit-learn), R
- Activities: Implement k-means and hierarchical clustering.

Lab 19: Factor Analysis

- Objective: Conduct factor analysis.
- Tools: Python (FactorAnalyzer), R
- Activities: Perform and interpret factor analysis.

Lab 20: Principal Component Analysis (PCA)

- Objective: Apply PCA for dimensionality reduction.
- Tools: Python (Scikit-learn), R
- Activities: Implement PCA and interpret the results.

Unit 7: Statistical Software Applications

Lab 21: Introduction to Statistical Software

- Objective: Get familiar with statistical software.
- Tools: R, SPSS, Python
- Activities: Basic operations and data management in R, SPSS, and Python.

Lab 22: Data Analysis Using R

- Objective: Perform data analysis using R.
- Tools: R
- Activities: Data manipulation, statistical analysis, and visualization in R.

Lab 23: Data Analysis Using SPSS

- Objective: Conduct data analysis using SPSS.
- Tools: SPSS
- Activities: Data entry, descriptive statistics, and inferential statistics in SPSS.

Lab 24: Data Analysis Using Python

- Objective: Perform data analysis using Python.

- Tools: Python (Pandas, NumPy, Matplotlib, SciPy)
- Activities: Data cleaning, preparation, and statistical modeling in Python.

Unit 8: Real-World Applications and Case Studies
Lab 25: Business Analytics
- Objective: Apply statistical techniques to business problems.
- Tools: Python, R, SPSS
- Activities: Analyze business datasets and derive insights.

Lab 26: Social Science Research
- Objective: Apply statistics in social science research.
- Tools: Python, R, SPSS
- Activities: Analyze social science data and interpret results.

Lab 27: Health Sciences
- Objective: Apply statistical methods to health science data.
- Tools: Python, R, SPSS
- Activities: Analyze health datasets and interpret findings.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
CO1	3	3	-	-	-	3	-	3
CO2	3	3	-	-	3	3	3	3
CO3	3	-	3	-	3	-	3	-
CO4	3	3	-	3	-	3	-	3
CO5	3	3	-	-	3	3	-	3

CO6	**3**	**3**	**3**	**-**	**3**	**-**	**3**	**3**
CO7	**3**	**3**	**3**	**-**	**3**	**3**	**3**	**-**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 301: MATHEMATICS FOR DATA SCIENCE
B.Tech. I Year I Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: High School Mathematics

Course Objective:

1. Introduce the basics of linear algebra and its importance in data science.
2. Develop an understanding of vectors, matrices, and their operations.
3. Cover the fundamental concepts of calculus that are applicable to data science.
4. Understand differentiation and integration in the context of data functions.
5. Introduce the fundamental principles of probability theory.
6. Apply probability concepts to data science problems.
7. Develop an understanding of descriptive and inferential statistics.
8. Apply statistical methods to analyze data.
9. Introduce discrete mathematics concepts relevant to data science.
10. Develop logical reasoning and problem-solving skills.
11. Introduce optimization techniques and their applications in data science.
12. Solve optimization problems using mathematical methods.

Learning Course Outcome:
After successful completion of the course, students will able to:

CO1: Apply Linear Algebra Techniques

- Understand and apply concepts of vectors, matrices, and linear transformations in data science problems.
- Compute eigenvalues and eigenvectors and use them in dimensionality reduction techniques like Principal Component Analysis (PCA).

CO2: Perform Calculus Operations

- Perform differentiation and integration of functions, and apply these techniques to optimize and analyze data-driven models.
- Utilize multivariable calculus to solve problems involving multiple variables and optimize complex functions.

CO3: Apply Probability Theory

- Understand and apply fundamental concepts of probability, including random variables, probability distributions, and expectation.
- Use probability theory to model uncertainty and variability in data.

CO4: Utilize Statistical Methods

- Employ descriptive statistics to summarize and visualize data effectively.
- Conduct inferential statistical analyses, including hypothesis testing and confidence interval estimation, to draw valid conclusions from data.

CO5: Implement Discrete Mathematics

- Apply concepts from set theory, logic, and combinatorics to solve problems in data science.
- Utilize discrete mathematics to understand and implement algorithms and data structures used in data processing.

CO6: Solve Optimization Problems

- Formulate and solve linear and nonlinear optimization problems relevant to data science.
- Use optimization techniques to improve machine learning models and enhance decision-making processes in data-driven applications.

Course Contents

UNIT-I	Linear Algebra	07 Hours

1. **Vectors and Spaces**
   - Definition of vectors and vector spaces
   - Linear combinations, span, and basis
   - Dot product and orthogonality
2. **Matrices and Matrix Operations**
   - Matrix addition, multiplication, and transposition
   - Identity and inverse matrices
   - Determinants and their properties
3. **Eigenvalues and Eigenvectors**
   - Definition and properties
   - Calculation of eigenvalues and eigenvectors
   - Applications in dimensionality reduction (e.g., PCA)

Lab/Practical:

- Implementation of matrix operations and eigenvalue computations using Python libraries (e.g., NumPy).

UNIT-II	Calculus	08 Hours

1. **Differentiation**
   - Limits and continuity
   - Derivatives of functions
   - Partial derivatives and gradients
2. **Integration**
   - Definite and indefinite integrals
   - Techniques of integration
   - Applications in probability and statistics
3. **Multivariable Calculus**
   - Functions of multiple variables
   - Gradient, divergence, and curl
   - Optimization: maxima, minima, and saddle points

Lab/Practical:

Practical examples of differentiation and integration using Python (e.g., SymPy for symbolic computation).

UNIT-III	Probability	08 Hours

1. **Basic Probability Concepts**
   - Sample spaces and events
   - Probability axioms and rules
   - Conditional probability and independence
2. **Random Variables**
   - Discrete and continuous random variables
   - Probability mass functions (PMF) and probability density functions (PDF)
   - Expectation, variance, and moments
3. **Common Probability Distributions**
   - Binomial, Poisson, and Normal distributions
   - Properties and applications
   - Central Limit Theorem

3.1 Basics of Probability
3.1.1 Definitions and Concepts
Experiment, Sample Space, and Events
Types of Events: Simple, Compound, Mutually Exclusive, Exhaustive
Basic Probability Principles: Classical, Relative Frequency, Subjective
3.1.2 Probability Axioms
Non-negativity
Normalization
Additivity
3.1.3 Calculation of Probabilities
Counting Techniques: Permutations and Combinations
Calculation of Probabilities Using Counting Methods
3.1.4 Examples and Applications
Real-world Examples Illustrating Basic Probability Concepts
3.2 Probability Rules and Theorems
3.2.1 Addition Rule
Addition Rule for Mutually Exclusive Events
General Addition Rule
3.2.2 Multiplication Rule
Multiplication Rule for Independent Events
General Multiplication Rule
3.2.3 Complementary Rule
Definition and Calculation
Applications of Complementary Rule

3.2.4 Bayes' Theorem
Statement and Interpretation
Applications of Bayes' Theorem in Real-world Scenarios
3.2.5 Law of Total Probability
Statement and Interpretation
Using the Law of Total Probability to Solve Problems
3.2.6 Examples and Applications
Practical Examples Using Probability Rules and Theorems
3.3 Conditional Probability and Independence
3.3.1 Conditional Probability
Definition and Calculation
Conditional Probability in Real-world Contexts
3.3.2 Independent Events
Definition and Identification
Properties of Independent Events
3.3.3 Dependent Events
Definition and Identification
Calculating Probability for Dependent Events
3.3.4 Applications and Examples
Solving Problems Involving Conditional Probability and Independence
Real-world Applications of Conditional Probability and Independence
3.4 Probability Distributions
3.4.1 Discrete Distributions
3.4.1.1 Binomial Distribution
Definition and Characteristics
Probability Mass Function (PMF)
Binomial Coefficient and Calculations
Mean and Variance of Binomial Distribution
Applications of Binomial Distribution
3.4.1.2 Poisson Distribution
Definition and Characteristics
Probability Mass Function (PMF)
Mean and Variance of Poisson Distribution
Applications of Poisson Distribution
Relationship Between Poisson and Binomial Distributions
3.4.2 Continuous Distributions
3.4.2.1 Normal Distribution
Definition and Characteristics
Probability Density Function (PDF)
Standard Normal Distribution and Z-scores
Empirical Rule (68-95-99.7 Rule)
Applications of Normal Distribution
3.4.2.2 Exponential Distribution
Definition and Characteristics
Probability Density Function (PDF)
Mean and Variance of Exponential Distribution
Memoryless Property

Applications of Exponential Distribution
3.4.3 Other Important Distributions
Overview of Uniform Distribution (Discrete and Continuous)
Overview of Gamma and Beta Distributions

Lab/Practical:

- Simulating random variables and probability distributions using Python (e.g., SciPy).

UNIT-IV	Statistics	07 Hours

1. **Descriptive Statistics**
    - Measures of central tendency: mean, median, mode
    - Measures of dispersion: range, variance, standard deviation
    - Data visualization: histograms, box plots
2. **Inferential Statistics**
    - Sampling methods and distributions
    - Hypothesis testing and p-values
    - Confidence intervals
3. **Regression Analysis**
    - Simple linear regression
    - Multiple linear regression
    - Assessing model fit: R-squared and residual analysis

Lab/Practical:

- Performing descriptive and inferential statistics using Python (e.g., Pandas, StatsModels).

UNIT-V	Discrete Mathematics	08 Hours

1. **Set Theory**
    - Sets, subsets, and power sets
    - Operations on sets: union, intersection, and difference
    - Venn diagrams
2. **Logic and Propositional Calculus**
    - Propositions and logical connectives
    - Truth tables and logical equivalence
    - Predicate logic and quantifiers
3. **Combinatorics**
    - Basic counting principles
    - Permutations and combinations
    - Applications in probability

Lab/Practical:

- Implementing set operations and combinatorial algorithms using Python.

UNIT-VI	Optimization	07 Hours

- **Optimization Basics**
  - Types of optimization problems: linear and nonlinear
  - Objective functions and constraints
  - Feasible regions and optimality conditions
- **Linear Programming**
  - Formulating linear programming problems
  - Graphical solution method
  - Simplex algorithm
- **Nonlinear Optimization**
  - Unconstrained optimization: gradient descent
  - Constrained optimization: Lagrange multipliers
  - Applications in machine learning (e.g., logistic regression)

Lab/Practical:

- Solving optimization problems using Python libraries (e.g., SciPy, PuLP).
- Implementation of matrix operations and eigenvalue computations using Python libraries (e.g., NumPy).
- Practical examples of differentiation and integration using Python (e.g., SymPy for symbolic computation).
- Simulating random variables and probability distributions using Python (e.g., SciPy).
- Performing descriptive and inferential statistics using Python (e.g., Pandas, StatsModels).
- Implementing set operations and combinatorial algorithms using Python.
- Solving optimization problems using Python libraries (e.g., SciPy, PuLP)

References
Textbooks

1. **Linear Algebra: "Linear Algebra and Its Applications" by David C. Lay, Steven R. Lay, and Judi J. McDonald**
   - A comprehensive introduction to linear algebra concepts and applications, with a focus on practical problem-solving techniques.
2. **Calculus: "Calculus: Early Transcendentals" by James Stewart**
   - An in-depth exploration of calculus concepts, including differentiation, integration, and multivariable calculus, with numerous examples and exercises.
3. **Probability: "Introduction to Probability" by Dimitri P. Bertsekas and John N. Tsitsiklis**
   - A foundational text on probability theory, covering both basic and advanced topics relevant to data science applications.
4. **Statistics: "Probability and Statistics for Engineers and Scientists" by Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, and Keying E. Ye**
   - A detailed guide to statistical methods, including descriptive and inferential statistics, with a focus on engineering and scientific applications.
5. **Discrete Mathematics: "Discrete Mathematics and Its Applications" by Kenneth H. Rosen**
   - A comprehensive introduction to discrete mathematics, covering set theory, logic, combinatorics, and graph theory.
6. **Optimization: "Introduction to Operations Research" by Frederick S. Hillier and Gerald J. Lieberman**
   - An authoritative text on optimization techniques, including linear programming and nonlinear optimization, with practical applications in various fields.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes/POs & PSOs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3	PSO4	PSO5
CO1: Apply Linear Algebra Techniques	3	3	-	2	3	-	-	3	2	2	-	-
CO2: Perform Calculus Operations	3	3	1	2	2	-	-	3	2	2	-	-
CO3: Apply Probability Theory	3	3	1	2	2	-	-	3	3	2	1	-
CO4: Utilize Statistical Methods	3	3	1	3	3	-	-	3	3	3	2	1
CO5: Implement Discrete Mathematics	3	3	-	2	2	-	-	3	2	2	1	-
CO6: Solve Optimization Problems	3	3	2	2	3	-	-	3	3	3	2	2

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 302: INTRODUCTION TO DATA ANALYTICS
B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Software Flow Basic

Course Objective:

- Introduce learners to the essential concepts, terminologies, and methodologies in data analytics, data science, and business analytics.
- Provide a comprehensive overview of the data analytics lifecycle, spanning data preparation, exploration, analysis, visualization, and decision-making processes.
- Emphasize the significance of data in various domains, industries, and modern business decision-making.
- Develop foundational skills in data analysis, interpretation, and the creation of actionable insights.
- Address the ethical considerations, biases, and responsibilities associated with handling and analyzing data.
- Expose students to popular tools, techniques, and platforms in the realms of data analytics and business intelligence.
- Explore real-world, industry-relevant use cases to bridge the gap between theory and application.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Define and articulate core concepts such as data analytics, data science, business analytics, big data, machine learning, and business intelligence.
- Navigate the entire data analytics lifecycle, from data acquisition to actionable insights presentation.
- Design and conceptualize basic data warehouses, appreciating the intricacies of multidimensional data representation.
- Utilize tools like Tableau, Excel, Python, and R for data visualization, analysis, and representation.
- Identify patterns, correlations, and trends in datasets, using both descriptive and inferential statistical methods.
- Understand the ethical dimensions, challenges, and biases inherent in data analysis and business decision-making.
- Evaluate the practical implications of data and business analytics across diverse industry scenarios.
- Differentiate and apply foundational techniques in data mining and machine learning in real-world contexts.

Course Contents

UNIT-I	Introduction to Data & Business Analytics	07 Hours

1.1 Definition and Evolution of Data Analytics
  1.1.1 Historical context
  1.1.2 Evolution of data analytics tools and methodologies

1.2 Emergence of Data Science and Business Analytics
1.2.1 Distinctive features of data science and business analytics
1.2.2 Interdisciplinary nature and industry applications
1.3 The Role of Data in Modern Decision-Making
1.3.1 Importance of data-driven decision-making
1.3.2 Impact on organizational strategies
1.4 Responsibilities and Ethics in Data Practices
1.4.1 Ethical considerations in data collection, handling, and dissemination
1.4.2 Professional responsibilities of data practitioners

UNIT-II	**Foundational Concepts**	**08 Hours**

2.1 Key Terminologies in Data Analytics
2.1.1 Definitions and distinctions: analytics, science, business analytics, big data, machine learning, business intelligence
2.2 Methodologies in Data Science and Business Analytics
2.2.1 Overview of methodologies: CRISP-DM, Agile, and others
2.2.2 Selecting appropriate methodologies for different scenarios
2.3 Understanding Big Data, Machine Learning, and Business Intelligence
2.3.1 Deep dive into big data technologies and challenges
2.3.2 Introduction to machine learning algorithms and applications
2.3.3 Business intelligence tools and their role in decision support

UNIT-III	**Data Analytics Lifecycle**	**08 Hours**

3.1 Introduction to the Stages of the Data Analytics Lifecycle
3.1.1 Defining the stages: acquisition, preparation, exploration, analysis, visualization, decision-making
3.2 Strategies and Techniques for Data Acquisition
3.2.1 Data collection methods: surveys, sensors, web scraping, etc.
3.2.2 Ethical considerations in data acquisition
3.3 Data Preparation: Cleaning and Transforming Raw Data
3.3.1 Techniques for data cleaning: handling missing values, outliers, and inconsistencies
3.3.2 Transforming data for analysis: normalization, encoding, scaling
3.4 Exploration, Analysis, and Visualization Techniques
3.4.1 Exploratory data analysis (EDA) methods
3.4.2 Advanced analysis techniques: regression, clustering, classification
3.4.3 Visualization best practices and storytelling with data
3.5 Effective Decision-Making Processes in Data Analytics
3.5.1 Incorporating analytics insights into decision-making
3.5.2 Decision-making frameworks and models

UNIT-IV	**Significance of Data in Business and Foundational Skills Development**	**07 Hours**

4.1 Examining Data's Impact Across Various Domains
4.1.1 Healthcare, finance, marketing, operations, and more
4.2 Case Studies: Industry Applications of Data Analytics
4.2.1 Real-world examples of successful data-driven initiatives
4.3 Modern Paradigms in Business Decision-Making with Data
4.3.1 Integrating data analytics into strategic planning

4.3.2 Case studies on transformative decision-making
5.1 Techniques for Data Analysis
5.1.1 Descriptive statistics: mean, median, mode
5.1.2 Inferential statistics: hypothesis testing, regression analysis
5.2 Interpreting Data: From Patterns to Actionable Insights
5.2.1 Identifying patterns, correlations, and trends
5.2.2 Turning insights into actionable business recommendations
5.3 Creating Actionable Insights for Business
5.3.1 Storytelling with data: effective communication of insights
5.3.2 Building dashboards for decision-makers

UNIT-V	Ethical Considerations in Data Analytics, Tools, Techniques, and Platforms	08 Hours

6.1 Unpacking Ethical Dimensions of Data Analysis
6.1.1 Privacy concerns and data anonymization
6.1.2 Ethical challenges in handling sensitive data
6.2 Addressing and Mitigating Biases in Data
6.2.1 Identifying and mitigating bias in algorithms and datasets
6.2.2 Ethical considerations in machine learning model development
6.3 Code of Ethics: Responsibilities of Data Practitioners
6.3.1 Ethical guidelines for data practitioners
6.3.2 Professional standards and certifications
7.1 Overview of Data Analytics Tools and Platforms
7.1.1 Comparative analysis of popular tools: Tableau, Excel, Python, R
7.1.2 Selection criteria for tools based on use cases
7.2 Hands-On Experience with Tableau, Excel, Python, and R
7.2.1 Practical exercises for each tool
7.2.2 Integrating tools for comprehensive analysis
7.3 Integration of Tools for Comprehensive Analysis
7.3.1 Developing end-to-end workflows for data analysis
7.3.2 Collaborative analytics: combining outputs from different tools

UNIT-VI	Real-World Use Cases	08 Hours

8.1 In-Depth Exploration of Industry-Relevant Use Cases
8.1.1 Case studies across diverse industries
8.1.2 Analyzing challenges and solutions in real-world scenarios
8.2 Bridging the Gap Between Theory and Application
8.2.1 Applying theoretical concepts to practical scenarios
8.2.2 Group projects and hands-on simulations
8.3 Analyzing and Solving Practical Scenarios through Case Studies
8.3.1 Problem-solving methodologies in analytics
8.3.2 Evaluating the success of implemented solutions.

REFERENCE BOOKS:
1. "Data Science for Business" by Foster Provost and Tom Fawcett
2. "Python for Data Analysis" by Wes McKinney
3. "Data Analytics Made Accessible" by Anil Maheshwari
4. "Data Science for Dummies" by Lillian Pierson
5. "Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani
6. "Storytelling with Data" by Cole Nussbaumer Knaflic
7. "Data Mining: Practical Machine Learning Tools and Techniques" by Ian H. Witten, Eibe Frank, and Mark A. Hall
8. "R for Data Science" by Hadley Wickham and Garrett Grolemund
9. "Big Data: A Revolution That Will Transform How We Live, Work, and Think" by Viktor Mayer-Schönberger and Kenneth Cukier
10. "Data Science from Scratch: First Principles with Python" by Joel Grus

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 303: ADVANCE DATA STRUCTURES

B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Programming for Problem Solving

Course Objective:

- Exploring basic data structures such as stacks and queues.
- Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
- Introduces sorting and pattern matching algorithms

Learning Course Outcome:

After successful completion of the course, students will able to:

- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using a variety of data structures, including hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and AVL-trees.

Course Contents

UNIT-I	Introduction to Data	07 Hours
Introduction to Data Structures, abstract data types, Linear list — singly linked list implementation, insertion, deletion and searching operations on linear list, Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations.		
UNIT-II	**Dictionaries**	**08 Hours**
linear list representation, skip list representation, operations - insertion, deletion and searching. Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, extendible hashing.		
UNIT-III	**Search Trees**	**08 Hours**
Binary Search Trees, Definition, Implementation, Operations- Searching, Insertion and Deletion, B- Trees, B+ Trees, AVL Trees, Definition, Height of an AVL Tree, Operations — Insertion, Deletion and Searching, Red –Black, Splay Trees.		
UNIT-IV	**Graphs**	**07 Hours**

Graph Implementation Methods. Graph Traversal Methods. Sorting: Quick Sort, Heap Sort, External Sorting- Model for external sorting, Merge Sort.		
UNIT-V	**Pattern Matching and Tries**	**08 Hours**
Pattern matching algorithms-Brute force, the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries, Suffix tries.		

TEXT BOOKS:

1. Fundamentals of Data Structures in C, 2 nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.

REFERENCE BOOK:

1. Data Structures: A Pseudocode Approach with C, 2 nd Edition, R. F. Gilberg and B.A.Forouzan, Cengage Learning.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 304: WEB FRAMEWORK

B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic understanding of Python programming.

Course Objective:

1. To understand the basics of Django and its applications in web development.
2. To learn to build dynamic web applications using the Django framework.
3. To integrate databases with Django applications.
4. To implement user authentication and authorization in Django.
5. To develop skills for testing and deploying Django applications.

Learning Course Outcome:

After successful completion of the course, students will able to:

CO1: Understand the architecture and components of the Django framework.

CO2: Develop dynamic web applications using Django.

CO3: Integrate and manage databases with Django ORM.

CO4: Implement user authentication and authorization.

CO5: Test and deploy Django applications.

Course Contents

UNIT-I	Overview of Django	07 Hours

History and evolution of Django, Features and benefits of using Django, Django architecture: MTV (Model-Template-View) pattern, **setting up Django:** Installation and configuration, Creating a Django project, Overview of Django project structure.

UNIT-II	Django Models and ORM	08 Hours

Introduction to Django Models: Creating and defining models, Model fields and field types, Database setup and configuration, **Django ORM:** QuerySets and querying the database, CRUD (Create, Read, Update, Delete) operations, Relationships between models (One-to-One, One-to-Many, Many-to-Many)

UNIT-III	Django Views and Templates	08 Hours

Django Views: Function-based views (FBV), Class-based views (CBV), Handling forms and form validation, **Django Templates:** Template language and syntax, Template inheritance and reusability, Context and context processors.

UNIT-IV	Django URLs and Routing	07 Hours
URL Configuration: URL patterns and routing, Including and referencing URLs from other apps, Named URL patterns and reverse resolution, **Dynamic URLs:** Capturing URL parameters, URL dispatching and redirection.		
UNIT-V	**User Authentication and Authorization**	**08 Hours**
Authentication System: User model and user management, Authentication views and forms, Password hashing and security, **Authorization:** Permissions and groups, Role-based access control, Custom user models and extending user profiles.		
UNIT-VI	**Advanced Django Topics**	**08 Hours**
Django Forms: Form classes and form handling, Model forms and form sets, Custom form fields and validation **Django Admin:** Customizing the admin interface, Adding models to the admin site, Custom admin actions and filters **Testing and Deployment:** Writing and running tests, Django testing framework, Deployment strategies (using WSGI, deploying on cloud platforms), Configuring static and media files		

REFERENCE BOOKS:

1. "Django for Beginners: Build websites with Python and Django" by William S. Vincent
2. "Two Scoops of Django 3.x: Best Practices for the Django Web Framework" by Audrey Roy Greenfeld and Daniel Roy Greenfeld
3. "Django 3 By Example: Build powerful and reliable Python web applications from scratch" by Antonio Mele
4. Official Django documentation: https://docs.djangoproject.com

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3	PSO4
CO1: Understand the architecture and components of the Django framework	3	2	-	-	3	-	-	2	-	2	-
CO2: Develop dynamic web applications using Django	3	3	2	-	3	-	-	3	2	3	-
CO3: Integrate and manage databases with Django ORM	3	3	2	-	3	-	-	3	3	3	2
CO4: Implement user authentication and authorization	3	3	-	2	3	-	-	2	3	3	3
CO5: Test and deploy Django applications	3	3	2	3	3	-	-	3	3	3	3
Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3	PSO4
CO1: Understand the architecture and components of the Django framework	3	2	-	-	3	-	-	2	-	2	-

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 305: COMPUTER ORGANIZATION AND ARCHITECTURE
B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: A Course on "Digital Electronics".

Course Objective:

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, microprogrammed control unit, pipelining and vector processing, memory organization and I/O systems, and multiprocessors

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand the basics of instruction sets and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
- Design a pipeline for consistent execution of instructions with minimum hazards.
- Recognize and manipulate representations of numbers stored in digital computers

Course Contents

UNIT-I	Digital Computers	07 Hours

Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.
Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.
Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT-II	Microprogrammed Control	08 Hours
Control memory, Address sequencing, micro program example, design of control unit. Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.		
UNIT-III	**Data Representation**	**08 Hours**
Data types, Complements, Fixed Point Representation, Floating Point Representation. Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating — point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.		
UNIT-IV	**Input-Output Organization**	**07 Hours**
Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access. Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.		
UNIT-V	**Reduced Instruction Set Computer**	**08 Hours**
CISC Characteristics, RISC Characteristics. Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor. Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache Coherence.		

TEXT BOOK:

1. Computer System Architecture – M. Morris Mano, Third Edition, Pearson/PHI.

REFERENCE BOOKS:

1. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, Vth Edition, McGraw Hill.
2. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
3. Structured Computer Organization – Andrew S. Tanenbaum, 4 th Edition, PHI/Pearson.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 303L: ADVANCE DATA STRUCTURES LAB
B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: A Course on "Programming for problem solving".

Course Objective:

- It covers various concepts of C programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Ability to develop C programs for computing and real-life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
- Ability to Implement searching and sorting algorithms

Course Contents

List of Experiments	07 Hours

1. Write a program that uses functions to perform the following operations on singly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on doubly linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linked list.:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implement stack (its operations) using
   i) Arrays ii) Pointers
5. Write a program that implement Queue (its operations) using
   i) Arrays ii) Pointers
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
   i) Quick sort ii) Heap sort iii) Merge sort
7. Write a program to implement the tree traversal methods (Recursive and Non Recursive).

8. Write a program to implement
   i) Binary Search tree ii) B Trees iii) B+ Trees iv) AVL trees
   v) Red - Black trees
9. Write a program to implement the graph traversal methods.
10. Implement a Pattern matching algorithms using Boyer- Moore, Knuth-Morris-Pratt

TEXT BOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

REFERENCE BOOK:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 304L: WEB FRAMEWORK LAB
B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Programming Lab Practice

Course Objective:

1. **Understand the Django Framework**.
2. **Model and Database Management.**
3. **Dynamic Web Development**.
4. **URL Routing and Navigation**.
5. **Form Handling and Validation**.
6. **User Authentication and Authorization**.
7. **Deployment of Django Applications**.
8. **Testing and Quality Assurance**.
9. **Integration with Third-Party Libraries and APIs**.
10. **Implementing Security Measures**.

Learning Course Outcome:

After successful completion of the course, students will able to:

CO1: Set up and configure the Django development environment.

CO2: Create and manage Django models and perform CRUD operations using Django ORM.

CO3: Develop dynamic web pages using Django views and templates.

CO4: Implement URL routing and navigation in Django applications.

CO5: Handle forms and perform validation in Django applications.

CO6: Implement user authentication and authorization in Django.

CO7: Deploy Django applications to live servers.

CO8: Write and run tests for Django applications to ensure quality and reliability.

CO9: Integrate third-party libraries and APIs with Django.

CO10: Implement security measures in Django applications.

Course Contents

Experiment Of List	07 Hours

Lab Assignment 1: Setting Up Django Environment

- **Objective**: Set up a Django development environment.
- **Tasks**:
    - Install Python and Django.
    - Create a new Django project.
    - Understand the project structure.
    - Run the development server.
- **Expected Outcome**: Successfully set up and run a Django project on a local server.

Lab Assignment 2: Creating Django Models

- **Objective**: Understand and create Django models.
- **Tasks**:
    - Define models for a sample application (e.g., Blog, E-commerce).
    - Migrate models to the database.
    - Use Django Admin to manage models.
- **Expected Outcome**: Successfully create and manage Django models.

Lab Assignment 3: Django ORM and Queries

- **Objective**: Perform CRUD operations using Django ORM.
- **Tasks**:
    - Insert, update, delete, and retrieve data using Django ORM.
    - Implement QuerySets and filtering.
    - Create relationships between models.
- **Expected Outcome**: Perform various database operations using Django ORM.

Lab Assignment 4: Building Views and Templates

- **Objective**: Develop views and templates for dynamic web pages.
- **Tasks**:
    - Create function-based views and class-based views.
    - Pass data to templates and render it.
    - Use template inheritance and filters.
- **Expected Outcome**: Develop dynamic web pages using Django views and templates.

Lab Assignment 5: URL Routing and Navigation

- **Objective**: Understand and implement URL routing in Django.
- **Tasks**:
    - Define URL patterns for different views.
    - Use named URL patterns and reverse resolution.
    - Implement navigation links in templates.
- **Expected Outcome**: Set up URL routing and navigation for a Django application.

Lab Assignment 6: Form Handling and Validation

- **Objective**: Handle forms and perform validation in Django.
- **Tasks**:
  - Create and process forms.
  - Implement form validation.
  - Use ModelForm for database operations.
- **Expected Outcome**: Successfully handle and validate forms in a Django application.

Lab Assignment 7: User Authentication and Authorization

- **Objective**: Implement user authentication and authorization.
- **Tasks**:
  - Set up user registration, login, and logout.
  - Implement password hashing and authentication.
  - Manage user permissions and access control.
- **Expected Outcome**: Implement a complete user authentication system in Django.

Lab Assignment 8: Integrating with Databases

- **Objective**: Integrate Django with various databases.
- **Tasks**:
  - Configure different databases (SQLite, PostgreSQL, MySQL).
  - Perform database migrations.
  - Use database-specific features in Django.
- **Expected Outcome**: Integrate and work with multiple databases in Django.

Lab Assignment 9: Deploying Django Applications

- **Objective**: Deploy a Django application to a live server.
- **Tasks**:
  - Prepare the application for deployment.
  - Set up a production environment (using WSGI, Gunicorn).
  - Deploy on cloud platforms (Heroku, AWS, etc.).
- **Expected Outcome**: Successfully deploy a Django application to a live server.

Lab Assignment 10: Testing Django Applications

- **Objective**: Write and run tests for Django applications.
- **Tasks**:
  - Write unit tests for models, views, and forms.
  - Use Django's testing framework.
  - Run tests and interpret results.
- **Expected Outcome**: Write and execute tests to ensure the quality of Django applications.

Lab Assignment 11: Integrating Third-Party Libraries

- **Objective**: Use third-party libraries and APIs in Django.
- **Tasks**:
    - Install and configure third-party libraries (e.g., Django Rest Framework, Celery).
    - Integrate external APIs.
    - Implement additional functionalities (e.g., social authentication, email services).
- **Expected Outcome**: Extend Django applications using third-party libraries and APIs.

Lab Assignment 12: Implementing Security Measures

- **Objective**: Implement security best practices in Django.
- **Tasks**:
    - Use Django's built-in security features (e.g., CSRF, XSS protection).
    - Implement SSL/TLS for secure communication.
    - Manage user sessions securely.
- **Expected Outcome**: Implement basic security measures to protect Django applications.

REFERENCE BOOK:

1. **"Django for Beginners" by William S. Vincent**: This book is an excellent starting point for beginners. It covers the basics of Django and takes you through building your first project.
    - ISBN-13: 978-1735467207
2. **"Two Scoops of Django 3.x: Best Practices for the Django Web Framework" by Daniel Roy Greenfeld and Audrey Roy Greenfeld**: This book provides best practices and tips for Django development, making it a valuable resource for intermediate to advanced users.
    - ISBN-13: 978-0692915721
3. **"Django Unleashed" by Andrew Pinkham**: A comprehensive guide that delves into more advanced topics in Django, including testing, deployment, and scaling applications.
    - ISBN-13: 978-0321985071
4. **"Django for APIs: Build web APIs with Python and Django" by William S. Vincent**: Focuses on building APIs with Django, which is essential for integrating with third-party services and creating backend systems.
    - ISBN-13: 978-1735467221

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance

  2. Assignments

  3. Case Study/Mini Project/Presentation

- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3	PSO4
CO1: Set up and configure the Django development environment	3	-	-	-	3	-	-	2	-	-	-
CO2: Create and manage Django models and perform CRUD operations using Django ORM	3	3	2	-	3	-	-	3	3	3	2
CO3: Develop dynamic web pages using Django views and templates	3	3	2	-	3	-	-	3	2	3	2
CO4: Implement URL routing and navigation in Django applications	3	2	2	-	3	-	-	2	2	2	2
CO5: Handle forms and perform validation in Django applications	3	3	2	-	3	-	-	3	2	3	2
CO6: Implement user authentication and authorization in Django	3	3	2	-	3	-	-	3	3	3	3
CO7: Deploy Django applications to live servers	3	3	2	-	3	-	-	3	3	3	2
CO8: Write and run tests for Django applications to ensure quality and reliability	3	3	2	3	3	-	-	3	3	3	2
CO9: Integrate third-party libraries and APIs with Django	3	2	2	-	3	-	-	3	3	3	2
CO10: Implement security measures in Django applications	3	3	2	-	3	-	-	3	3	3	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 306L: DATA VISUALIZATION - R PROGRAMMING/ POWER BI
B.Tech. II Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: 10 +2 Math Statistics

Course Objective:

- Effective use of Business Intelligence (BI) technology (Tableau) to apply data visualization
- To discern patterns and relationships in the data.
- To build Dashboard applications.
- To communicate the results clearly and concisely.
- To be able to work with different formats of data sets.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand How to import data into Tableau.
- Understand Tableau concepts of Dimensions and Measures.
- Develop Programs and understand how to map Visual Layouts and Graphical Properties.
- Create a Dashboard that links multiple visualizations.
- Use graphical user interfaces to create Frames for providing solutions to real world
- problems.

Course Contents

UNIT-I	Lab Problems	07 Hours

1. Understanding Data, What is data, where to find data, Foundations for building Data Visualizations, Creating Your First visualization?

2. Getting started with Tableau Software using Data file formats, connecting your Data to Tableau, creating basic charts(line, bar charts, Tree maps),Using the Show me panel.

3. Tableau Calculations, Overview of SUM, AVR, and Aggregate features, Creating custom calculations and fields.

4. Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.

5. Editing and Formatting Axes, Manipulating Data in Tableau data, Pivoting Tableau data.

6. Structuring your data, Sorting and filtering Tableau data, Pivoting Tableau data.

7. Advanced Visualization Tools: Using Filters, Using the Detail panel, using the Size panels, customizing filters, Using and Customizing tooltips, Formatting your data with colors.

8. Creating Dashboards & Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing & Publishing your Visualization.

9. Tableau file types, publishing to Tableau Online, Sharing your visualizations, printing, and Exporting.

10. Creating custom charts, cyclical data and circular area charts, Dual Axis charts.

REFERENCE BOOKS:

1. Microsoft Power BI cookbook, Brett Powell, 2nd edition.
2. R Programming for Data Science by Roger D. Peng (References)
3. The Art of R Programming by Norman Matloff Cengage Learning India.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO’s-PO’s & PSO’s MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 401: DISCRETE MATHEMATICS

B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: 10 +2 discrete mathematics

Course Objective:

- Introduces elementary discrete mathematics for computer science and engineering.
- Topics include formal logic notation, methods of proof, induction, sets, relations, algebraic structures, elementary graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand and construct precise mathematical proofs
- Apply logic and set theory to formulate precise statements
- Analyze and solve counting problems on finite and discrete structures
- Describe and manipulate sequences
- Apply graph theory in solving computing problems

Course Contents

UNIT-I	Foundations of Discrete Mathematics	07 Hours

Sets, subsets, and set operations, Logic and propositions: Propositional logic, logical connectives, truth tables, tautologies, contradictions, Predicate logic: Quantifiers, predicates, and logical equivalences, Proof techniques: Direct proof, proof by contradiction, mathematical induction

Hands-on Exercises:

- Set operations and Venn diagrams
- Constructing truth tables and logical equivalences
- Writing proofs using different proof techniques

UNIT-II	Combinatorics and Counting	08 Hours

Basic counting principles: Addition and multiplication rules, Permutations and combinations, Binomial theorem and Pascal's triangle, Pigeonhole principle, Inclusion-exclusion principle

Hands-on Exercises:

- Solving combinatorial problems with permutations and combinations
- Applying the binomial theorem to data science problems
- Using inclusion-exclusion principle in real-world scenarios

UNIT-III	Graph Theory	08 Hours

Introduction to graphs: Definitions, terminology, and types of graphs, Graph representation: Adjacency matrix, adjacency list, Graph traversal: Breadth-first search (BFS), depth-first search (DFS), Shortest path algorithms:

Dijkstra's algorithm, Bellman-Ford algorithm, Graph applications in data science: Social network analysis, clustering

Hands-on Exercises:

- Implementing BFS and DFS in Python
- Solving shortest path problems with Dijkstra's and Bellman-Ford algorithms
- Analyzing social networks using graph theory

UNIT-IV	Relations and Functions	07 Hours

Relations: Properties, representation, equivalence relations, partial orderings, Functions: Definitions, types of functions, composition, and inverses, Recurrence relations: Solving linear recurrence relations, applications in algorithm analysis

Hands-on Exercises:

- Working with relations and their properties
- Implementing and analyzing functions in Python
- Solving and analyzing recurrence relations

UNIT-V	Number Theory and Cryptography	08 Hours

Divisibility and prime numbers, Congruences and modular arithmetic, Euclidean algorithm and applications, Basics of cryptography: RSA algorithm, public-key

Hands-on Exercises:

- Solving problems involving divisibility and modular arithmetic
- Implementing the Euclidean algorithm in Python
- Encrypting and decrypting messages using the RSA algorithm

UNIT-VI	Probability and Statistics	08 Hours

Basic probability theory: Sample spaces, events, probability axioms, Conditional probability and Bayes' theorem, Random variables and probability distributions, Expectation, variance, and standard deviation, Applications in data science: Hypothesis testing, regression analysis

Hands-on Exercises:

1. Calculating probabilities and applying Bayes' theorem
2. Working with random variables and probability distributions
3. Performing hypothesis testing and regression analysis in Python

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-Hill, 1st ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe l. Mott, Abraham Kandel, Teodore P. Baker, Prentis Hall of India, 2nd ed.

REFERENCE BOOKS:

3. "Discrete Mathematics and Its Applications" by Kenneth H. Rosen
4. "Discrete Mathematics with Applications" by Susanna S. Epp
5. "Graph Theory with Applications" by Bondy and Murty
6. "Introduction to the Theory of Computation" by Michael Sipser
7. "Probability and Statistics for Engineers and Scientists" by Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, and Keying Ye
8. "Cryptography and Network Security: Principles and Practice" by William Stallings

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4
CO1	3	3	3	-	-	3	-	3	-
CO2	3	3	3	-	-	3	-	3	3
CO3	3	3	-	-	3	3	-	-	3
CO4	3	3	-	3	-	-	3	-	-
CO5	3	3	-	-	3	3	-	3	-
CO6	3	3	-	3	-	-	3	3	3

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 402: INTRODUCTION TO ARTIFICIAL INTELLIGENCE
B.Tech. II Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Knowledge on Data Structures.

Course Objective:

- To understand the historical development and key milestones in the field of AI.
- To explore logical and probabilistic approaches to AI.
- To learn the principles and applications of evolutionary algorithms.
- To gain knowledge of fundamental and advanced machine learning techniques.
- To comprehend the structure and functioning of neural networks and deep learning models.
- To understand the basics of natural language processing and its applications.
- To explore generative models and their real-world applications.
- To discuss the ethical implications and considerations in AI.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Describe the historical context and evolution of AI.
- Apply logical and probabilistic reasoning techniques in AI.
- Implement evolutionary algorithms for optimization problems.
- Develop and evaluate machine learning models using supervised and unsupervised learning techniques.
- Construct and train neural networks and deep learning models for various applications.
- Apply natural language processing techniques to analyze and generate text.
- Utilize generative models such as GANs and VAEs for image generation and data augmentation.
- Assess and address ethical issues related to bias, fairness, and misuse of AI technologies.

Course Contents

UNIT-I	Introduction to Artificial Intelligence (AI)	07 Hours

1.1: Historical Development of AI
1.1.1: Early AI Concepts
1.1.1.1: The Turing Test
1.1.1.2: Dartmouth Workshop (1956)
1.1.2: Key Milestones in AI Development
1.1.2.1: McCarthy's Dartmouth Proposal
1.1.2.2: Birth of the First AI Programs (1950s)
1.1.3: Influential AI Contributors

1.1.3.1: Alan Turing
1.1.3.2: John McCarthy
1.2: Logical Approach to AI
1.2.1: Propositional and First-Order Logic
1.2.1.1: Boolean Algebra
1.2.1.2: Predicate Logic
1.2.2: Knowledge Representation
1.2.2.1: Semantic Networks
1.2.2.2: Frames and Ontologies
1.2.3: Rule-Based Systems
1.2.3.1: Production Rules
1.2.3.2: Forward and Backward Chaining
1.3: Knowledge-Based Systems
1.3.1: Expert Systems
1.3.1.1: Rule-Based Expert Systems
1.3.1.2: Knowledge Engineering
1.3.2: Inference Engines
1.3.2.1: Forward Chaining Inference
1.3.2.2: Backward Chaining Inference
1.3.3: Case-Based Reasoning
1.3.3.1: Retrieval and Adaptation
1.3.3.2: Case Base Maintenance

UNIT-II	Probabilistic Approaches to AI	08 Hours

2.1: Probability Concepts in AI
2.1.1: Probability Distributions
2.1.1.1: Discrete and Continuous Distributions
2.1.1.2: Bayes' Theorem
2.1.2: Bayesian Inference
2.1.2.1: Prior, Likelihood, and Posterior
2.1.2.2: Markov Chain Monte Carlo (MCMC)
2.2: Bayesian Networks
2.2.1: Structure and Representation
2.2.1.1: Directed Acyclic Graphs (DAGs)
2.2.1.2: Conditional Probability Tables (CPTs)
2.2.2: Inference in Bayesian Networks
2.2.2.1: Exact Inference (Variable Elimination)
2.2.2.2: Approximate Inference (Monte Carlo Methods)

UNIT-III	Evolutionary Intelligence	08 Hours

3.1: Evolutionary Algorithms 3.1.1: Genetic Algorithms 3.1.1.1: Selection Operators 3.1.1.2: Crossover and Mutation 3.1.2: Evolutionary Strategies 3.1.2.1: Self-Adaptation 3.1.2.2: Evolution Strategies (ES) 3.2: Applications of Evolutionary Intelligence 3.2.1: Optimization Problems 3.2.1.1: Traveling Salesman Problem (TSP) 3.2.1.2: Genetic Programming for Symbolic Regression 3.2.2: Genetic Programming 3.2.2.1: Tree-Based Representations 3.2.2.2: Evolution of Computer Programs		
UNIT-IV	**Introduction to Machine Learning (ML)**	**07 Hours**
4.1: Basics of Machine Learning 4.1.1: Types of Learning (Supervised, Unsupervised, Reinforcement) 4.1.1.1: Labeling Data 4.1.1.2: Rewards and Policies 4.1.2: Model Evaluation Metrics 4.1.2.1: Accuracy, Precision, Recall 4.1.2.2: F1-Score, ROC-AUC 4.2: Supervised Learning 4.2.1: Linear Regression 4.2.1.1: Ordinary Least Squares (OLS) 4.2.1.2: Gradient Descent 4.2.2: Logistic Regression 4.2.2.1: Binary Classification 4.2.2.2: Multinomial Logistic Regression 4.2.3: Decision Trees and Random Forests 4.2.3.1: Tree Splitting Criteria 4.2.3.2: Ensemble Learning 4.3: Unsupervised Learning 4.3.1: Clustering Algorithms (K-Means, Hierarchical) 4.3.1.1: Cluster Initialization 4.3.1.2: Agglomerative Clustering 4.3.2: Principal Component Analysis (PCA) 4.3.2.1: Dimensionality Reduction 4.3.2.2: Eigenvalue Decomposition		
UNIT-V	**Neural Networks and Deep Learning**	**08 Hours**

5.1: Neural Networks Fundamentals
5.1.1: Perceptrons and Activation Functions
5.1.1.1: McCulloch-Pitts Neuron
5.1.1.2: Sigmoid and ReLU Activation
5.1.2: Feedforward Neural Networks
5.1.2.1: Feedforward Propagation
5.1.2.2: Backpropagation
5.2: Deep Learning
5.2.1: Convolutional Neural Networks (CNNs)
5.2.1.1: Convolution and Pooling Layers
5.2.1.2: Object Detection
5.2.2: Recurrent Neural Networks (RNNs)
5.2.2.1: Long Short-Term Memory (LSTM)
5.2.2.2: Sequence Generation
5.2.3: Transfer Learning
5.2.3.1: Fine-Tuning Pretrained Models
5.2.3.2: Domain Adaptation

UNIT-VI	Natural Language Processing (NLP) and Generative Intelligence	08 Hours

6.1: NLP Basics
6.1.1: Tokenization and Text Preprocessing
6.1.1.1: Token Segmentation
6.1.1.2: Stopword Removal
6.1.2: Word Embeddings (Word2Vec, GloVe)
6.1.2.1: Word Vectors
6.1.2.2: Vector Similarity
6.2: NLP Applications
6.2.1: Sentiment Analysis
6.2.1.1: Text Classification
6.2.1.2: Sentiment Lexicons
6.2.2: Named Entity Recognition
6.2.2.1: Named Entity Types
6.2.2.2: Sequence Labeling
6.2.3: Text Generation
6.2.3.1: Recurrent Text Generation
6.2.3.2: Transformer-Based Models

7.1: Introduction to Generative Models
7.1.1: Generative Adversarial Networks (GANs)
7.1.1.1: Generator and Discriminator
7.1.1.2: GAN Training Process
7.1.2: Variational Autoencoders (VAEs)
7.1.2.1: Encoder and Decoder Networks

7.1.2.2: Latent Variable Sampling
7.2: Real-World Uses and Ethical Considerations
7.2.1: Applications of Generative Models
7.2.1.1: Image Generation
7.2.1.2: Data Augmentation
7.2.2: Ethical Implications of Generative AI
7.2.2.1: Bias and Fairness
7.2.2.2: Deepfakes and Misuse

TEXT BOOK:

1. Artificial Intelligence: A Modern Approach, Third Edition, Stuart Russell and Peter Norvig, Pearson Education.

REFERENCE BOOKS:

1. "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig
2. "Pattern Recognition and Machine Learning" by Christopher M. Bishop
3. "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville
4. "Probabilistic Graphical Models: Principles and Techniques" by Daphne Koller and Nir Friedman
5. "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto
6. "Natural Language Processing in Action" by Lane, Howard, and Hapke
7. "Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play" by David Foster

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 403: DATABASE MANAGEMENT SYSTEMS

B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: A course on "Data Structures".

Course Objective:

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Gain knowledge of fundamentals of DBMS, database design and normal forms
- Master the basics of SQL for retrieval and management of data.
- Be acquainted with the basics of transaction processing and concurrency control.
- Familiarity with database storage structures and access techniques

Course Contents

UNIT-I	Database System Applications	07 Hours

A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model

UNIT-II	Introduction to the Relational Model	08 Hours

Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views.
Relational Algebra, Tuple relational Calculus, Domain relational calculus.

UNIT-III	SQL	08 Hours

QUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.

Schema Refinement: Problems caused by redundancy, decompositions, problems related to

decomposition, reasoning about functional dependencies, First, Second, Third normal forms, BCNF, lossless join decomposition, multivalued dependencies, Fourth normal form, Fifth normal form.		
UNIT-IV	**Transaction Property**	**07 Hours**
Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions.		
UNIT-V	**File Organization**	**08 Hours**
Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree based Indexing, Comparison of File Organizations, Indexes- Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.		

TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 404: OPERATING SYSTEMS
B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

1. **Course Prerequisites:** A course on “Computer Programming and Data Structures”.
2. A course on “Computer Organization and Architecture”.

Course Objective:

- Introduce operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
- Introduce the issues to be considered in the design and development of operating system
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

Learning Course Outcome:
After successful completion of the course, students will able to:

- Will be able to control access to a computer and the files that may be shared
- Demonstrate the knowledge of the components of computers and their respective roles in computing.
- Ability to recognize and resolve user problems with standard operating environments.
- Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

Course Contents

UNIT-I	Operating System - Introduction	07 Hours
Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls. **Process** - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads		
UNIT-II	**CPU Scheduling**	**08 Hours**
Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec **Deadlocks** - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock		

UNIT-III	Process Management and Synchronization	08 Hours
The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors **Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.		
UNIT-IV	**Memory Management and Virtual Memory**	**07 Hours**
Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.		
UNIT-V	**File System Interface and Operations**	**08 Hours**
Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.		

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Advanced programming in the UNIX environment, W.R. Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems- Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System A Design Approach- Crowley, TMH.
3. Modern Operating Systems, Andrew S. Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/ Pearson Education
5. UNIX Internals -The New Frontiers, U. Vahalia, Pearson Education.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 403L: DATABASE MANAGEMENT SYSTEMS LAB
B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: "Database Management Systems".

Course Objective:

- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

Learning Course Outcome:
After successful completion of the course, students will able to:

- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

Course Contents

List of Experiments	07 Hours

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A. Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.)
   B. Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

TEXT BOOKS:

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill, 3rd Edition
2. Database System Concepts, Silberschatz, Korth, McGraw Hill, V edition.

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C.J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 404L: OPERATING SYSTEMS LAB
B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: A course on "Programming for Problem Solving", A course on "Computer Organization and Architecture".
Co-requisite: A course on "Operating Systems".

Course Objective:

- To provide an understanding of the design aspects of operating system concepts through simulation
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

Learning Course Outcome:
After successful completion of the course, students will able to:

- Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management.
- Able to implement C programs using Unix system calls

Course Contents

List of Experiments	07 Hours

1. Write C programs to simulate the following CPU Scheduling algorithms a) FCFS b) SJF c) Round Robin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer — Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs c) Message Queues
d) Shared Memory
6. Write C programs to simulate the following memory management techniques a) Paging b) Segmentation
7. Write C programs to simulate Page replacement policies a) FCFS b) LRU c) Optimal

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems – Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System - A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI
4. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education
5. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 405L: NODE JS/ REACT JS
B.Tech. II Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Object Oriented Programming through Java, HTML Basics

Course Objective:

- To implement the static web pages using HTML and do client side validation using JavaScript.
- To design and work with databases using Java
- To develop an end to end application using java full stack.
- To introduce Node JS implementation for server side programming.
- To experiment with single page application development using React.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Build a custom website with HTML, CSS, and Bootstrap and little JavaScript.
- Demonstrate Advanced features of JavaScript and learn about JDBC
- Develop Server – side implementation using Java technologies like
- Develop the server – side implementation using Node JS.
- Design a Single Page Application using React.

Course Contents

Exercises	07 Hours

1. Build a responsive web application for shopping cart with registration, login, catalog and cart pages using CSS3 features, flex and grid.
2. Make the above web application responsive web application using Bootstrap framework.
3. Use JavaScript for doing client – side validation of the pages implemented in experiment 1 and experiment 2.
4. Explore the features of ES6 like arrow functions, callbacks, promises, async/await. Implement an application for reading the weather information from openweathermap.org and display the information in the form of a graph on the web page.
5. Develop a java stand alone application that connects with the database (Oracle / mySql) and perform the CRUD operation on the database tables.
6. Create an xml for the bookstore. Validate the same using both DTD and XSD.
7. Design a controller with servlet that provides the interaction with application developed in experiment 1 and the database created in experiment 5.

8. Maintaining the transactional history of any user is very important. Explore the various session tracking mechanism (Cookies, HTTP Session)
9. Create a custom server using http module and explore the other modules of Node JS like OS, path, event.
10. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
11. For the above application create authorized end points using JWT (JSON Web Token).
12. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages.
13. Create a service in react that fetches the weather information from openweathermap.org and the display the current and historical weather information using graphical representation using chart.js
14. Create a TODO application in react with necessary components and deploy it into github.

REFERENCE BOOKS:

1. Jon Duckett, Beginning HTML, XHTML, CSS, and JavaScript, Wrox Publications, 2010
2. Bryan Basham, Kathy Sierra and Bert Bates, Head First Servlets and JSP, O'Reilly Media, 2nd Edition, 2008.
3. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, A Press.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 501 DESIGN AND ANALYSIS OF ALGORITHMS
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites:

- A course on “Computer Programming and Data Structures”.
- A course on “Advanced Data Structures”.

Course Objective:

- Introduces the notations for analysis of the performance of algorithms and the data structure of disjoint sets.
- Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming, greedy, branch and bound methods) and mention problems for which each technique is appropriate
- Describes how to evaluate and compare different algorithms using worst-, average-, and best case analysis.
- Explains the difference between tractable and intractable problems, and introduces the problems that are P, NP and NP complete.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Analyze the performance of algorithms
- Choose appropriate data structures and algorithm design methods for a specified application
- Understand the choice of data structures and the algorithm design methods

Course Contents

	Introduction	07 Hours
Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations- Big oh notation, Omega notation, Theta notation and Little oh notation. **Divide and conquer:** General method, applications-Binary search, Quick sort, Merge sort, Strassen’s matrix multiplication.		
UNIT-II	**Disjoint Sets**	**08 Hours**
Disjoint set operations, union and find algorithms, Priority Queue- Heaps, Heapsort **Backtracking:** General method, applications, n-queen’s problem, sum of subsets problem, graph Coloring, hamitonian cycles.		
UNIT-III	**Dynamic Programming**	**08 Hours**

General method, applications- Optimal binary search tree, 0/1 knapsack problem, All pairs shortest path problem, Traveling salesperson problem, Reliability design.

UNIT-IV	Greedy method	07 Hours

General method, applications-Job sequencing with deadlines, knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

Basic Traversal and Search Techniques: Techniques for Binary Trees, Techniques for Graphs, Connected components, Biconnected components.

UNIT-V	Branch and Bound	08 Hours

General method, applications - Traveling salesperson problem, 0/1 knapsack problem - LC Branch and Bound solution, FIFO Branch and Bound solution.

NP-Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP-Hard and NP-Complete classes, Cook's theorem.

TEXT BOOK:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharan, University press, 1998.

REFERENCE BOOKS:

1. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.
2. Introduction to Algorithms, second edition, T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, PHI Pvt. Ltd./ Pearson Education.
3. Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R. Tamassia, John Wiley and sons.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 502: R PROGRAMMING
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Introduction To Data Science

Course Objective:

1. To introduce students to the R programming environment.
2. To teach data manipulation, visualization, and analysis using R.
3. To cover statistical modeling and machine learning techniques in R.
4. To enable students to perform reproducible research using R.
5. To familiarize students with advanced topics such as parallel computing and package development in R.

Learning Course Outcome:
After successful completion of the course, students will able to:

CO1: Understand the basic concepts and environment of R programming.
CO2: Perform data manipulation and cleaning using R packages.
CO3: Create and customize various types of data visualizations.
CO4: Conduct statistical analysis and interpret the results.
CO5: Apply machine learning algorithms to real-world datasets.
CO6: Utilize advanced R programming techniques including parallel computing and package development.
CO7: Conduct reproducible research and document analysis using RMarkdown.

Course Contents

UNIT-I	Introduction to R	07 Hours
Overview of R and RStudio, Installing R and RStudio, Basic R Syntax, Variables and Data Types, Vectors, Lists, Matrices, Arrays, and Data Frames, Basic Operations in R		
UNIT-II	**Data Manipulation in R**	**08 Hours**
Importing and Exporting Data, Data Cleaning and Preparation, dplyr and tidyr for Data Manipulation, String Manipulation with stringr, Handling Missing Values		

UNIT-III	Data Visualization in R	08 Hours
Introduction to Data Visualization, Basic Plotting with Base R, Advanced Plotting with ggplot2, Customizing Plots, Creating Interactive Visualizations with plotly and shiny		
UNIT-IV	**Statistical Analysis in R**	**07 Hours**
Descriptive Statistics, Probability Distributions, Hypothesis Testing, ANOVA and Regression Analysis, Time Series Analysis.		
UNIT-V	**Machine Learning in R**	**08 Hours**
Introduction to Machine Learning, Supervised Learning: Regression and Classification, Unsupervised Learning: Clustering and Association Rules, Model Evaluation and Tuning, Using caret and mlr Packages for Machine Learning.		
UNIT-V	**Advanced Topics in R**	**08 Hours**
Introduction to Parallel Computing in R, Writing Functions and Scripts, Package Development in R, Version Control with Git and GitHub, Reproducible Research with RMarkdown		

- Introduction to Data Visualization
- Basic Plotting with Base R
- Advanced Plotting with ggplot2
- Customizing Plots
- Creating Interactive Visualizations with plotly and shiny

REFERENCE BOOKS:

1. "R for Data Science" by Hadley Wickham and Garrett Grolemund ISBN-13: 978-1491910399
2. "Advanced R" by Hadley Wickham ISBN-13: 978-1498759809
3. "The Art of R Programming" by Norman Matloff ISBN-13: 978-1593273842
4. "R in Action" by Robert I. Kabacoff ISBN-13: 978-1617291388

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students’ reading and writing

skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Course Outcomes (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1: Understand the basic concepts and environment of R programming	3	-	-	-	2	-	-	2	-	-
CO2: Perform data manipulation and cleaning using R packages	3	3	2	2	3	-	-	3	3	3
CO3: Create and customize various types of data visualizations	3	3	2	-	3	-	-	3	2	3
CO4: Conduct statistical analysis and interpret the results	3	3	2	2	3	-	-	3	3	3
CO5: Apply machine learning algorithms to real-world datasets	3	3	2	2	3	-	-	3	3	3
CO6: Utilize	3	3	2	2	3	-	-	3	3	3

advanced R programming techniques including parallel computing and package development										
CO7: Conduct reproducible research and document analysis using RMarkdown	**3**	**3**	**2**	**2**	**3**	**-**	**-**	**3**	**3**	**3**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 503: BUSINESS ECONOMICS AND FINANCIAL ANALYSIS
B.Tech. III Year I Sem.

Teaching Scheme: TH: - 2 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

To learn the basic Business types, impact of the Economy on Business and Firms specifically.

To analyze the Business from the Financial Perspective.

Learning Course Outcome:

After successful completion of the course, students will able to:

The students will understand th e various Forms of Business and the impact of economic variables on the Business. The Demand, Supply, Production, Cost, Market Structure, Pricing aspects are learnt. The Students can study the firm's financial position by analysing the Financial Statements of a Company.

Course Contents

UNIT-I	Introduction to Business and Economics	07 Hours

Business: Structure of Business Firm, Theory of Firm, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company, Non-Conventional Sources of Finance.

Economics: Significance of Economics, Micro and Macro Economic Concepts, Concepts and Importance of National Income, Inflation, Money Supply in Inflation, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

UNIT-II	Demand and Supply Analysis	08 Hours

Elasticity of Demand: Elasticity, Types of Elasticity, Law of Demand, Measurement and Significance of Elasticity of Demand, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making, Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting.

Supply Analysis: Determinants of Supply, Supply Function & Law of Supply.

UNIT-III	Production, Cost, Market Structures & Pricing	08 Hours

Production Analysis: Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production Functions.
Cost analysis: Types of Costs, Short run and Long run Cost Functions.
Market Structures: Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, Monopolistic Competition.
Pricing: Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis, Cost Volume Profit Analysis.

UNIT-IV	Financial Accounting	07 Hours

Accounting concepts and Conventions, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, Preparation of Final Accounts.

UNIT-V	Financial Analysis through Ratios	08 Hours

Concept of Ratio Analysis, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems).
Introduction to Fund Flow and Cash Flow Analysis (simple problems).

TEXT BOOKS:

1. D.D. Chaturvedi, S.L. Gupta, Business Economics - Theory and Applications, International Book House Pvt. Ltd. 2013.
2. Dhanesh K Khatri, Financial Accounting, Tata McGraw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd. 2012.

REFERENCE BOOKS:

1. Paresh Shah, Financial Accounting for Management 2e, Oxford Press, 2015.
2. S.N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, Vikas Publications, 2013.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 504: DESCRIPTIVE ANALYTICS
B.Tech III Year I Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: The introduction of the Advanced English Communication

Course Objective:

- To understand the role and importance of descriptive analytics and business intelligence in decision-making.
- To familiarize students with business intelligence architecture and decision support science.
- To learn methods for data collection, data cleaning, and preparation for analysis.
- To develop skills in exploratory data analysis and data visualization.
- To teach students how to design and develop reports, dashboards, and scorecards for data visualization.
- To apply statistical techniques and advanced data mining approaches for summarizing and interpreting data.
- To enable students to build and manage BI projects effectively.
- To enhance problem-solving skills by analyzing real-world datasets.
- To communicate findings effectively through data storytelling and visualization.
- To equip students with the skills necessary to measure BI success and value.
- To prepare students for careers in business intelligence, analytics, and decision support systems.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Explain the principles and significance of descriptive analytics and business intelligence.
- Understand the architecture and functional areas of business intelligence tools.
- Collect, clean, and prepare data for analysis.
- Perform exploratory data analysis to identify patterns and trends.
- Use statistical methods and advanced data mining techniques to describe and summarize data.
- Create effective data visualizations, reports, dashboards, and scorecards to communicate insights.
- Utilize analytics tools such as Excel, Python, R, Power BI, and Tableau.
- Design and implement BI projects using industry-standard methodologies and tools.
- Analyze real-world datasets and derive actionable insights.
- Evaluate the effectiveness and value of business intelligence initiatives.
- Present data findings clearly and persuasively to stakeholders.
- Apply decision support techniques to analyze and explore data for decision-making.

Course Contents

UNIT-I	Introduction to Descriptive Analytics and Business Intelligence, Business Intelligence Architecture and Decision Support Science	07 Hours

1.1 Overview of Descriptive Analytics
- 1.1.1 Definition and Scope of Descriptive Analytics
  - Understanding Descriptive Analytics
  - Differentiating Descriptive Analytics from Predictive and Prescriptive Analytics
  - The role of Descriptive Analytics in the Analytics Spectrum
- 1.1.2 Key Concepts and Terminology
  - Data, Information, and Knowledge
  - Metrics and Measures
  - Key Performance Indicators (KPIs)
- 1.1.3 Historical Development of Descriptive Analytics
  - Evolution from Traditional Reporting to Modern Analytics
  - Major Milestones in the Development of Descriptive Analytics
- 1.1.4 Tools and Technologies
  - Overview of Popular Descriptive Analytics Tools (Excel, Tableau, Power BI)
  - Introduction to Programming Languages Used in Descriptive Analytics (Python, R)

1.2 Importance of Business Intelligence
- 1.2.1 Definition and Scope of Business Intelligence (BI)
  - Understanding Business Intelligence
  - Components of BI Systems
- 1.2.2 Benefits of Business Intelligence
  - Enhancing Decision-Making
  - Improving Operational Efficiency
  - Identifying Business Opportunities
  - Supporting Strategic Initiatives
- 1.2.3 BI in Various Industries
  - Case Studies: Use of BI in Healthcare, Retail, Finance, and Manufacturing
  - Industry-Specific Applications of BI
- 1.2.4 Integration of Descriptive Analytics in BI
  - How Descriptive Analytics Fits within the BI Framework
  - The Role of Descriptive Analytics in BI Workflows

1.3 Role in Decision-Making Processes
- 1.3.1 Types of Decisions Supported by BI
  - Strategic Decisions
  - Tactical Decisions
  - Operational Decisions
- 1.3.2 Decision-Making Models and Theories
  - Rational Decision-Making Model
  - Data-Driven Decision-Making
  - The Role of Intuition in Decision-Making
- 1.3.3 Data-Driven Decision-Making Process
  - Steps in the Data-Driven Decision-Making Process
  - Collecting and Preparing Data
  - Analyzing Data and Generating Insights
  - Making Decisions Based on Data Insights
- 1.3.4 Case Studies and Examples
  - Real-World Examples of BI Supporting Decision-Making
  - Success Stories from Various Organizations

1.3.5 Challenges and Limitations
Common Challenges in Implementing BI and Descriptive Analytics
Addressing Data Quality Issues
Overcoming Resistance to Data-Driven Decision-Making
2.1 Components of BI Architecture
2.1.1 Introduction to BI Architecture
Definition and Purpose of BI Architecture
Overview of BI Systems and Their Importance
2.1.2 Data Sources and Data Integration
Types of Data Sources (Structured, Unstructured, Semi-Structured)
ETL (Extract, Transform, Load) Processes
Data Warehousing
Characteristics of Data Warehouses
Differences Between Data Warehouses and Databases
2.1.3 Data Storage and Management
Data Marts
Data Lakes
Data Models (Star Schema, Snowflake Schema)
2.1.4 Analytical Processing
Online Analytical Processing (OLAP)
OLAP Operations (Slice, Dice, Drill Down, Roll Up)
OLAP Types (MOLAP, ROLAP, HOLAP)
Real-Time Analytics and Streaming Data
2.1.5 BI Tools and Technologies
Overview of BI Tools (Reporting Tools, Dashboards, Scorecards)
Advanced BI Tools (AI-Powered BI, Predictive Analytics Integration)
2.1.6 Security and Governance in BI
Data Security Best Practices
Data Governance Frameworks
Compliance and Regulatory Considerations
2.2 Decision Support Systems
2.2.1 Overview of Decision Support Systems (DSS)
Definition and Importance of DSS
Historical Development of DSS
2.2.2 Components of DSS
Database Management System (DBMS)
Model-Based Management System (MBMS)
User Interface
2.2.3 Types of Decision Support Systems
Data-Driven DSS
Model-Driven DSS
Knowledge-Driven DSS
Document-Driven DSS
Communication-Driven DSS
2.2.4 Decision Support Techniques
Simulation and Modeling
What-If Analysis

Sensitivity Analysis
2.2.5 Integration of DSS with BI
How DSS Enhances BI Capabilities
Case Studies of DSS Applications in BI
2.2.6 Trends and Innovations in DSS
Emerging Technologies in DSS (AI, Machine Learning)
Future Directions and Innovations
2.3 BI Project Lifecycle
2.3.1 Introduction to BI Project Lifecycle
Importance of Project Management in BI
Overview of BI Project Stages
2.3.2 Planning and Requirements Gathering
Defining Project Objectives and Scope
Stakeholder Analysis and Involvement
Requirements Elicitation Techniques
2.3.3 Designing BI Solutions
BI System Design Principles
Architecture Design
Data Modeling and Schema Design
2.3.4 Implementation and Development
Setting Up Data Integration and ETL Processes
Development of BI Reports and Dashboards
User Interface and Experience Design
2.3.5 Testing and Quality Assurance
Types of Testing (Unit Testing, Integration Testing, System Testing)
Ensuring Data Quality and Accuracy
User Acceptance Testing (UAT)
2.3.6 Deployment and Maintenance
Deployment Strategies (Phased, Parallel, Big Bang)
Training and Support for Users
Ongoing Maintenance and Optimization
2.3.7 Evaluating BI Project Success
Key Performance Indicators (KPIs) for Project Evaluation
Measuring Return on Investment (ROI)
Continuous Improvement and Feedback Loops.

UNIT-II	Data Collection and Preparation and Exploratory Data Analysis (EDA)	08 Hours

3.1 Data Sources and Types
3.1.1 Introduction to Data Sources
Definition and Importance of Data Sources
Overview of Data Sources in Business Intelligence
3.1.2 Structured Data
Characteristics and Examples of Structured Data
Sources of Structured Data (Databases, Spreadsheets)
Use Cases in Business Analytics
3.1.3 Unstructured Data

Characteristics and Examples of Unstructured Data
Sources of Unstructured Data (Text, Images, Videos)
Challenges and Opportunities with Unstructured Data
3.1.4 Semi-Structured Data
Characteristics and Examples of Semi-Structured Data
Sources of Semi-Structured Data (XML, JSON)
Use Cases and Integration with Structured Data
3.1.5 Internal vs. External Data Sources
Internal Data Sources (Enterprise Systems, CRM, ERP)
External Data Sources (Social Media, Public Data, Market Data)
Combining Internal and External Data for Comprehensive Analysis
3.2 Data Collection Methods
3.2.1 Overview of Data Collection
Importance of Effective Data Collection
Ethical Considerations in Data Collection
3.2.2 Manual Data Collection
Techniques for Manual Data Collection (Surveys, Interviews)
Pros and Cons of Manual Data Collection
3.2.3 Automated Data Collection
Techniques for Automated Data Collection (Web Scraping, APIs)
Tools and Technologies for Automated Data Collection
Benefits and Challenges of Automation
3.2.4 Real-Time Data Collection
Introduction to Real-Time Data Collection
Methods for Collecting Real-Time Data (Sensors, IoT Devices)
Applications and Use Cases for Real-Time Data
3.2.5 Data Collection in Business Intelligence
Best Practices for Data Collection in BI Projects
Ensuring Data Quality and Integrity During Collection
Legal and Compliance Issues in Data Collection
3.3 Data Cleaning Techniques
3.3.1 Introduction to Data Cleaning
Importance of Data Cleaning in Data Preparation
Common Issues Encountered in Raw Data
3.3.2 Identifying and Correcting Data Errors
Types of Data Errors (Typos, Inconsistencies, Duplicates)
Methods for Identifying Errors (Automated Tools, Manual Review)
Techniques for Correcting Errors (Standardization, Validation)
3.3.3 Data Transformation
Normalization and Standardization of Data
Data Formatting and Conversion
Aggregation and Summarization of Data
3.3.4 Dealing with Inconsistent Data
Identifying Inconsistent Data (Date Formats, Units of Measure)
Methods for Resolving Inconsistencies (Standard Rules, Algorithms)
Tools for Data Cleaning and Transformation (Python, R, Excel)
3.4 Handling Missing Data and Outliers

3.4.1 Introduction to Missing Data
Causes and Types of Missing Data (MCAR, MAR, MNAR)
Impact of Missing Data on Analysis
3.4.2 Techniques for Handling Missing Data
Deletion Methods (Listwise, Pairwise)
Imputation Methods (Mean/Median Imputation, Regression Imputation, Multiple Imputation)
Advanced Techniques (KNN Imputation, Machine Learning-Based Imputation)
3.4.3 Introduction to Outliers
Definition and Identification of Outliers
Causes and Impact of Outliers on Data Analysis
3.4.4 Techniques for Handling Outliers
Detection Methods (Z-Score, IQR, Visualization Techniques)
Treatment Methods (Capping, Transformation, Removal)
Considerations for Keeping or Removing Outliers Based on Context
3.4.5 Tools and Software for Data Cleaning and Preparation
Overview of Popular Data Cleaning Tools (OpenRefine, Trifacta, DataWrangler)
Using Python for Data Cleaning (Pandas, NumPy)
Using R for Data Cleaning (dplyr, tidyr)
4.1 Introduction to EDA
4.1.1 Definition and Importance of EDA
What is Exploratory Data Analysis?
The Role of EDA in the Data Analysis Process
Benefits of Conducting EDA
4.1.2 Objectives of EDA
Uncovering Patterns
Spotting Anomalies
Testing Hypotheses
Checking Assumptions
4.1.3 Steps in the EDA Process
Initial Data Inspection
Data Cleaning and Preparation
Data Visualization and Summary Statistics
Identifying Patterns, Trends, and Relationships
4.1.4 Tools for EDA
Software and Tools Used for EDA (Python, R, Excel, Tableau)
Introduction to Python Libraries (Pandas, Matplotlib, Seaborn)
Introduction to R Libraries (ggplot2, dplyr)
4.2 Descriptive Statistics
4.2.1 Overview of Descriptive Statistics
Definition and Importance
Types of Descriptive Statistics
4.2.2 Measures of Central Tendency
Mean
Definition and Calculation
Advantages and Disadvantages
Applications in Business Analytics
Median

Definition and Calculation
Advantages and Disadvantages
Applications in Business Analytics
Mode
Definition and Calculation
Advantages and Disadvantages
Applications in Business Analytics
4.2.3 Measures of Dispersion
Range
Definition and Calculation
Interpretation
Variance
Definition and Calculation
Interpretation
Standard Deviation
Definition and Calculation
Interpretation
Applications of Dispersion Measures in Business Analytics
4.2.4 Measures of Shape
Skewness
Definition and Calculation
Interpretation and Applications
Kurtosis
Definition and Calculation
Interpretation and Applications
4.2.5 Visualizing Descriptive Statistics
Histograms
Construction and Interpretation
Use Cases
Boxplots
Construction and Interpretation
Use Cases
Bar Charts
Construction and Interpretation
Use Cases
Best Practices for Visualizing Summary Statistics
4.3 Identifying Patterns and Trends
4.3.1 Time Series Analysis
Introduction to Time Series Data
Definition and Importance
Examples of Time Series Data in Business
Identifying Trends
Techniques for Identifying Trends
Visualization Techniques (Line Charts, Moving Averages)
Seasonality and Cycles
Identifying Seasonal Patterns
Techniques for Seasonality Analysis

Visualization Techniques (Seasonal Decomposition)
Applications of Time Series Analysis in Business
4.3.2 Pattern Recognition Techniques
Clustering
K-Means Clustering
Algorithm and Steps
Advantages and Disadvantages
Practical Applications
Hierarchical Clustering
Algorithm and Steps
Advantages and Disadvantages
Practical Applications
Classification
Decision Trees
Algorithm and Steps
Advantages and Disadvantages
Practical Applications
Support Vector Machines (SVM)
Algorithm and Steps
Advantages and Disadvantages
Practical Applications
Identifying Patterns in Data
Using Visualization Techniques (Heatmaps, Scatter Plots)
Applications of Pattern Recognition in Business
4.3.3 Anomaly Detection
Definition and Importance of Anomaly Detection
Methods for Detecting Anomalies
Z-Score Method
Calculation and Interpretation
Practical Applications
Interquartile Range (IQR) Method
Calculation and Interpretation
Practical Applications
Machine Learning Techniques for Anomaly Detection
Overview of Algorithms (Isolation Forest, LOF)
Practical Applications
Applications of Anomaly Detection in Business (Fraud Detection, Quality Control)
4.4 Correlation Analysis
4.4.1 Introduction to Correlation
Definition and Importance of Correlation
Types of Correlation
Positive Correlation
Negative Correlation
No Correlation
4.4.2 Calculating Correlation Coefficients
Pearson Correlation Coefficient
Definition and Calculation

Interpretation and Applications
Spearman Rank Correlation
Definition and Calculation
Interpretation and Applications
Kendall Tau Correlation
Definition and Calculation
Interpretation and Applications
4.4.3 Interpreting Correlation Results
Strength and Direction of Correlation
Interpretation of Correlation Coefficients
Practical Applications in Business
Correlation vs. Causation
Understanding the Difference
Common Misconceptions
Practical Applications of Correlation Analysis
Identifying Relationships Between Variables
Use Cases in Business Analytics
4.4.4 Visualizing Correlations
Correlation Matrices
Construction and Interpretation
Practical Applications
Scatter Plot Matrices
Construction and Interpretation
Practical Applications
Heatmaps for Correlation
Construction and Interpretation
Practical Applications

UNIT-III	**Statistical Methods for Data Summarization and Advanced Data Mining Techniques**	**08 Hours**

5.1 Measures of Central Tendency
5.1.1 Overview of Central Tendency
Definition and Importance in Data Analysis
Applications in Business Intelligence
5.1.2 Mean (Arithmetic Mean)
Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.1.3 Median
Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.1.4 Mode

Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.1.5 Comparing Measures of Central Tendency
When to Use Mean, Median, and Mode
Impact of Outliers on Central Tendency
Case Studies and Practical Applications
5.2 Measures of Dispersion and Variability
5.2.1 Overview of Dispersion and Variability
Definition and Importance in Data Analysis
Applications in Business Intelligence
5.2.2 Range
Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.2.3 Variance
Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.2.4 Standard Deviation
Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.2.5 Interquartile Range (IQR)
Definition and Formula
Calculation Methods
Advantages and Disadvantages
Practical Examples and Applications
5.2.6 Comparing Measures of Dispersion
When to Use Range, Variance, Standard Deviation, and IQR
Impact of Outliers on Dispersion Measures
Case Studies and Practical Applications
5.3 Probability Distributions
5.3.1 Introduction to Probability Distributions
Definition and Importance in Data Analysis
Types of Probability Distributions
5.3.2 Discrete Probability Distributions
Binomial Distribution
Definition, Formula, and Properties
Practical Examples and Applications
Poisson Distribution
Definition, Formula, and Properties
Practical Examples and Applications

5.3.3 Continuous Probability Distributions
Normal Distribution
Definition, Formula, and Properties
Practical Examples and Applications
Exponential Distribution
Definition, Formula, and Properties
Practical Examples and Applications
5.3.4 Understanding and Using Probability Distributions
Fitting Data to Distributions
Using Probability Distributions for Predictions
Case Studies and Practical Applications
5.4 Hypothesis Testing
5.4.1 Introduction to Hypothesis Testing
Definition and Importance in Data Analysis
Types of Hypotheses: Null and Alternative
5.4.2 Steps in Hypothesis Testing
Formulating Hypotheses
Choosing the Appropriate Test
Setting the Significance Level (Alpha)
Calculating Test Statistic and P-Value
Making a Decision: Reject or Fail to Reject the Null Hypothesis
5.4.3 Types of Hypothesis Tests
Z-Test
Definition, Formula, and When to Use
Practical Examples and Applications
T-Test
One-Sample T-Test
Definition, Formula, and When to Use
Practical Examples and Applications
Two-Sample T-Test (Independent and Paired)
Definition, Formula, and When to Use
Practical Examples and Applications
Chi-Square Test
Definition, Formula, and When to Use
Practical Examples and Applications
ANOVA (Analysis of Variance)
Definition, Formula, and When to Use
Practical Examples and Applications
5.4.4 Interpreting Hypothesis Test Results
Understanding P-Values and Confidence Intervals
Making Business Decisions Based on Hypothesis Test Results
Case Studies and Practical Applications
6.1 Introduction to Data Mining
6.1.1 Overview of Data Mining
Definition and Importance
History and Evolution of Data Mining
Applications in Various Industries

6.1.2 Data Mining Process
Steps in the Data Mining Process: Data Selection, Data Preprocessing, Data Transformation, Data Mining, Interpretation/Evaluation
Data Mining Methodologies: CRISP-DM, SEMMA, KDD
6.1.3 Key Concepts in Data Mining
Data Warehousing and Data Marts
Data Mining vs. Machine Learning
Ethical Considerations in Data Mining
6.2 Clustering and Classification
6.2.1 Clustering Techniques
Definition and Importance
Types of Clustering: Hard vs. Soft Clustering
Common Clustering Algorithms
K-Means Clustering
Algorithm Overview
Choosing the Number of Clusters (K)
Practical Examples and Applications
Hierarchical Clustering
Algorithm Overview
Agglomerative vs. Divisive Methods
Dendrogram Analysis
DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
Algorithm Overview
Identifying Core Points and Clusters
Handling Noise in Data
6.2.2 Classification Techniques
Definition and Importance
Types of Classification: Binary, Multiclass, Multilabel
Common Classification Algorithms
Decision Trees
Algorithm Overview
Splitting Criteria (Gini, Entropy)
Pruning Techniques
Practical Examples and Applications
Random Forest
Ensemble Method Overview
Building and Interpreting Random Forest Models
Support Vector Machines (SVM)
Algorithm Overview
Kernel Functions
Practical Examples and Applications
Naive Bayes
Algorithm Overview
Assumptions and Practical Use Cases
Neural Networks for Classification
Overview of Perceptrons and Multilayer Networks
Training Neural Networks for Classification Tasks

6.2.3 Model Evaluation and Validation
Confusion Matrix and Performance Metrics (Accuracy, Precision, Recall, F1-Score)
Cross-Validation Techniques
Handling Imbalanced Data
6.3 Association Rule Mining
6.3.1 Introduction to Association Rule Mining
Definition and Importance
Applications in Market Basket Analysis, Recommender Systems
6.3.2 Key Concepts in Association Rule Mining
Support, Confidence, and Lift
Apriori Algorithm
Algorithm Overview
Finding Frequent Itemsets
Generating Association Rules
Practical Examples and Applications
Eclat Algorithm
Algorithm Overview
Depth-First Search Approach
Practical Examples and Applications
FP-Growth (Frequent Pattern Growth)
Algorithm Overview
Building and Using FP-Trees
Practical Examples and Applications
6.3.3 Advanced Topics in Association Rule Mining
Mining Multidimensional Association Rules
Handling Categorical and Continuous Data
Pruning and Improving Rule Quality
6.4 Anomaly Detection
6.4.1 Introduction to Anomaly Detection
Definition and Importance
Applications in Fraud Detection, Network Security, Fault Detection
6.4.2 Techniques for Anomaly Detection
Statistical Methods
Z-Score and Thresholding
Box Plot Analysis
Practical Examples and Applications
Machine Learning Methods
Clustering-Based Methods (e.g., K-Means for Anomaly Detection)
Classification-Based Methods (e.g., One-Class SVM, Isolation Forest)
Neural Networks for Anomaly Detection
Autoencoders
LSTM for Sequence Anomalies
6.4.3 Evaluation of Anomaly Detection Methods
Precision, Recall, and F1-Score for Anomaly Detection
ROC and AUC Analysis
Handling Imbalanced Data and Rare Events

UNIT-IV	Data Visualization and Business Intelligence Tools	07 Hours

7.1 Principles of Data Visualization
- 7.1.1 Introduction to Data Visualization
  - Definition and Importance
  - History and Evolution of Data Visualization
  - Applications in Various Industries
- 7.1.2 Key Principles of Effective Data Visualization
  - Clarity and Simplicity
  - Accuracy and Integrity
  - Efficiency and Speed
  - Aesthetics and Design
- 7.1.3 Understanding the Audience
  - Identifying Audience Needs
  - Tailoring Visualizations to Different Stakeholders
  - Ensuring Accessibility and Inclusivity
- 7.1.4 Data Visualization Techniques
  - Comparison of Different Visualization Techniques
  - Choosing the Right Visualization for Different Data Types

7.2 Creating Effective Charts and Graphs
- 7.2.1 Types of Charts and Graphs
  - Bar Charts
    - Vertical and Horizontal Bar Charts
    - Stacked and Grouped Bar Charts
  - Line Charts
    - Simple Line Charts
    - Multiple Line Charts
  - Pie Charts
    - When and How to Use Pie Charts
    - Alternatives to Pie Charts
  - Scatter Plots
    - Correlation and Trend Analysis
    - Bubble Charts
  - Histograms and Box Plots
    - Distribution and Variability Analysis
  - Heatmaps
    - Visualizing Data Density
    - Using Color Scales Effectively
- 7.2.2 Best Practices for Creating Charts and Graphs
  - Choosing Appropriate Scales and Axes
  - Labeling and Annotating Effectively
  - Using Color and Style Consistently

Avoiding Common Pitfalls (e.g., Misleading Scales, Overcomplication)
7.2.3 Advanced Charting Techniques
Dual-Axis Charts
Combination Charts
Small Multiples
Interactive and Dynamic Charts
7.3 Dashboards and Scorecards
7.3.1 Introduction to Dashboards
Definition and Purpose
Types of Dashboards (Strategic, Analytical, Operational)
7.3.2 Designing Effective Dashboards
Key Elements of a Dashboard
Principles of Good Dashboard Design
Layout and Composition
Visual Hierarchy
Interactivity and User Experience
7.3.3 Tools for Creating Dashboards
Overview of Popular Tools (Tableau, Power BI, Excel, Google Data Studio)
Features and Capabilities of Each Tool
Choosing the Right Tool for the Job
7.3.4 Building and Implementing Dashboards
Step-by-Step Guide to Building a Dashboard
Connecting to Data Sources
Creating Interactive Elements (Filters, Drill-Downs)
Testing and Validating Dashboards
Deploying and Sharing Dashboards
7.3.5 Scorecards for Performance Monitoring
Definition and Purpose
Key Metrics and KPIs
Designing Effective Scorecards
Integrating Scorecards with Dashboards
7.4 Data Storytelling
7.4.1 The Art of Data Storytelling
Definition and Importance
Elements of a Good Data Story
The Role of Narrative in Data Visualization
7.4.2 Structuring a Data Story
Identifying the Core Message
Building the Narrative Arc (Introduction, Rising Action, Climax, Conclusion)
Supporting the Story with Data
7.4.3 Techniques for Effective Data Storytelling
Using Visuals to Enhance the Narrative
Combining Text and Visuals
Engaging the Audience Emotionally and Intellectually
7.4.4 Case Studies in Data Storytelling
Analyzing Successful Data Stories
Lessons Learned from Real-World Examples

7.4.5 Tools and Technologies for Data Storytelling
Overview of Tools (e.g., PowerPoint, Tableau Story Points, Flourish)
Integrating Different Media (Text, Visuals, Audio, Video)
Tips for Using Tools Effectively

8.1 Excel for Data Analysis

8.1.1 Introduction to Excel for BI
Overview of Excel's Capabilities in Data Analysis
Differences between Excel and other BI Tools

8.1.2 Data Management in Excel
Data Entry and Data Cleaning
Using Functions and Formulas
Data Validation and Conditional Formatting

8.1.3 Descriptive Statistics and Analysis
Applying Basic Statistical Functions (SUM, AVERAGE, COUNT, etc.)
Using Advanced Statistical Functions (STDEV, VAR, etc.)

8.1.4 PivotTables and PivotCharts
Creating and Customizing PivotTables
Using PivotCharts for Data Visualization
Filtering and Sorting Data in PivotTables

8.1.5 Advanced Excel Techniques
VLOOKUP, HLOOKUP, and INDEX-MATCH Functions
Using Excel Tables and Data Models
Introduction to Power Query and Power Pivot

8.1.6 Visualization and Reporting
Creating Charts and Graphs
Customizing Chart Elements
Building Interactive Dashboards in Excel

8.2 Python (pandas, matplotlib, seaborn)

8.2.1 Introduction to Python for Data Analysis
Overview of Python's Role in Data Analysis
Setting Up the Python Environment (Anaconda, Jupyter Notebooks)

8.2.2 Data Manipulation with pandas
Introduction to pandas Data Structures (Series, DataFrame)
Data Importing and Exporting
Data Cleaning and Preparation (Handling Missing Values, Duplicates)
Data Transformation (Filtering, Sorting, Grouping, Merging)

8.2.3 Data Visualization with matplotlib
Basics of matplotlib
Creating Basic Plots (Line, Bar, Scatter)
Customizing Plots (Labels, Titles, Legends)

8.2.4 Advanced Visualization with seaborn
Introduction to seaborn and Its Features
Creating Statistical Graphics (Histograms, KDE Plots, Pair Plots)
Customizing Seaborn Plots

8.2.5 Integrating pandas with matplotlib and seaborn
Combining Data Manipulation and Visualization
Creating Complex Visualizations for EDA

8.2.6 Real-World Data Analysis Projects
Hands-On Projects Using Real Datasets
Best Practices for Code and Data Management
8.3 R for Statistical Computing
8.3.1 Introduction to R and RStudio
Overview of R's Capabilities for Statistical Analysis
Setting Up R and RStudio
8.3.2 Data Management in R
Importing and Exporting Data
Data Cleaning and Preparation
Data Transformation with dplyr
8.3.3 Statistical Analysis with R
Descriptive Statistics (Summary, Mean, Median, Mode)
Inferential Statistics (t-tests, ANOVA)
Correlation and Regression Analysis
8.3.4 Data Visualization with ggplot2
Introduction to ggplot2
Creating Basic Plots (Histograms, Scatter Plots, Box Plots)
Customizing Plots with Themes and Labels
8.3.5 Advanced Visualization Techniques
Faceting and Grouping Data in Plots
Creating Interactive Visualizations with Shiny
8.3.6 Real-World Data Analysis Projects
Hands-On Projects Using R
Applying R to Business Intelligence Scenarios
8.4 Power BI and Tableau
8.4.1 Introduction to Power BI and Tableau
Overview of Power BI and Tableau Capabilities
Differences and Use Cases for Each Tool
8.4.2 Data Import and Preparation
Connecting to Data Sources (Excel, SQL, Web, etc.)
Data Cleaning and Transformation
Creating Data Models
8.4.3 Building Visualizations in Power BI
Creating Basic Visuals (Charts, Tables, Cards)
Customizing Visuals with Power BI Features
Using DAX for Advanced Calculations
8.4.4 Building Visualizations in Tableau
Creating Basic Visuals (Bar, Line, Pie Charts)
Customizing Visuals with Tableau Features
Using Calculated Fields and Table Calculations
8.4.5 Dashboards and Reports
Designing Interactive Dashboards
Best Practices for Dashboard Layout and Design
Sharing and Publishing Dashboards
8.4.6 Advanced Analytics and Insights
Using Power BI and Tableau for Advanced Data Analysis

	Applying Machine Learning Models in BI Tools Real-World Business Intelligence Projects	
UNIT-V	**Building and Managing BI Projects and Real-World Applications and Case Studies**	**08 Hours**

9.1 BI Project Planning and Execution
- 9.1.1 Introduction to BI Projects
  - Understanding the Scope and Objectives of BI Projects
  - Key Elements of a Successful BI Project
- 9.1.2 Project Planning and Management
  - Setting Project Goals and Milestones
  - Developing a Project Timeline
  - Allocating Resources and Responsibilities
  - Risk Management and Mitigation Strategies
- 9.1.3 Data Requirements and Integration
  - Identifying Data Sources and Requirements
  - Data Integration Techniques and Best Practices
  - Ensuring Data Quality and Consistency
- 9.1.4 Agile Methodology in BI Projects
  - Overview of Agile Methodology
  - Applying Agile Practices to BI Projects
  - Sprints, Backlogs, and Iterative Development
- 9.1.5 Collaboration and Communication
  - Building Effective Project Teams
  - Stakeholder Engagement and Management
  - Communication Strategies and Tools

9.2 Developing Reports and Dashboards
- 9.2.1 Introduction to Reporting in BI
  - Purpose and Importance of BI Reports
  - Types of BI Reports (Operational, Analytical, Strategic)
- 9.2.2 Designing Effective Reports
  - Principles of Effective Report Design
  - Structuring Reports for Clarity and Impact
  - Using KPIs and Metrics in Reports
- 9.2.3 Dashboard Design and Development
  - Introduction to Dashboards
  - Best Practices for Dashboard Design
  - Interactive Dashboards and User Experience
- 9.2.4 Tools for Report and Dashboard Development
  - Overview of BI Reporting Tools (Power BI, Tableau, etc.)
  - Step-by-Step Guide to Creating Reports and Dashboards
  - Advanced Features and Customization
- 9.2.5 Real-World Reporting and Dashboard Projects
  - Case Studies and Examples of Effective BI Reports
  - Hands-On Projects: Creating Reports and Dashboards

Peer Reviews and Feedback
9.3 Case Studies of BI Projects
9.3.1 Overview of Case Studies
Importance of Learning from Real-World Examples
Selection Criteria for Case Studies
9.3.2 Successful BI Project Case Studies
Case Study 1: Implementing a Sales Dashboard
Project Background and Objectives
Data Integration and Analysis
Challenges and Solutions
Outcomes and Impact
Case Study 2: Enhancing Customer Insights with BI
Project Background and Objectives
Data Collection and Cleaning
Analysis Techniques and Tools
Outcomes and Impact
9.3.3 Lessons Learned from BI Projects
Common Challenges in BI Projects
Best Practices and Key Takeaways
Applying Lessons to Future Projects
9.4 Measuring BI Success and Value
9.4.1 Importance of Measuring BI Success
Why Measuring Success is Crucial
Key Metrics for Evaluating BI Projects
9.4.2 Developing a BI Success Framework
Defining Success Criteria
Establishing KPIs and Metrics
Creating a Measurement Plan
9.4.3 Quantitative Measures of BI Success
Financial Metrics (ROI, Cost Savings)
Performance Metrics (Efficiency, Accuracy)
User Adoption and Satisfaction
9.4.4 Qualitative Measures of BI Success
User Feedback and Surveys
Case Study Reviews and Success Stories
Impact on Decision-Making and Strategy
9.4.5 Continuous Improvement in BI
Monitoring and Reporting on BI Performance
Identifying Areas for Improvement
Implementing Changes and Measuring Impact
10.1 Industry Applications of Descriptive Analytics
10.1.1 Introduction to Industry Applications
Overview of Descriptive Analytics Across Industries
Importance of Domain Knowledge in Analytics
10.1.2 Descriptive Analytics in Retail
Sales Analysis and Customer Segmentation
Inventory Management and Demand Forecasting

Case Study: Analyzing Sales Data for a Retail Chain
10.1.3 Descriptive Analytics in Healthcare
Patient Data Analysis and Healthcare Outcomes
Resource Allocation and Utilization
Case Study: Improving Patient Care Through Data Analysis
10.1.4 Descriptive Analytics in Finance
Fraud Detection and Risk Management
Portfolio Performance Analysis
Case Study: Risk Assessment and Mitigation in Banking
10.1.5 Descriptive Analytics in Marketing
Campaign Performance and Customer Insights
Market Basket Analysis and Customer Loyalty
Case Study: Enhancing Marketing Strategies with Analytics
10.1.6 Descriptive Analytics in Manufacturing
Quality Control and Process Improvement
Predictive Maintenance and Operational Efficiency
Case Study: Optimizing Production Processes Through Data
10.2 Hands-on Projects with Real-World Datasets
10.2.1 Project Planning and Setup
Defining Project Goals and Objectives
Selecting Appropriate Datasets
Setting Up Analytical Tools and Environment
10.2.2 Data Collection and Cleaning
Gathering Data from Multiple Sources
Data Preprocessing Techniques
Ensuring Data Quality and Integrity
10.2.3 Exploratory Data Analysis (EDA)
Applying Descriptive Statistics
Visualizing Data to Identify Trends and Patterns
Using EDA to Formulate Hypotheses
10.2.4 Advanced Data Analysis Techniques
Applying Statistical Methods and Data Mining Techniques
Clustering and Classification for Segmentation
Association Rule Mining and Anomaly Detection
10.2.5 Data Visualization and Reporting
Creating Charts, Graphs, and Dashboards
Developing Reports to Communicate Findings
Best Practices for Effective Data Storytelling
10.2.6 Presenting Project Results
Preparing a Presentation Outline
Communicating Insights to Stakeholders
Receiving and Incorporating Feedback
10.3 Analyzing and Interpreting Data-Driven Insights
10.3.1 Principles of Data Interpretation
Understanding Data Context and Relevance
Differentiating Between Correlation and Causation
10.3.2 Data-Driven Decision-Making

Translating Data Insights into Business Strategies Case Study: Using Data to Drive Business Decisions 10.3.3 Communicating Insights to Non-Technical Audiences Simplifying Complex Data for Stakeholder Understanding Techniques for Effective Data Storytelling 10.3.4 Evaluating the Impact of Data Insights Measuring the Effectiveness of Data-Driven Strategies Case Study: Assessing the Impact of Data Insights on Business Performance 10.3.5 Continuous Improvement Through Data Analysis Using Feedback Loops to Enhance Analytical Models Adapting Strategies Based on New Data Insights Case Study: Iterative Improvement in Business Analytics		
UNIT-VI	**Communication and Data Storytelling, Ethics and Best Practices in Business Intelligence**	**07 Hours**
11.1 Techniques for Effective Data Storytelling 11.1.1 Introduction to Data Storytelling Definition and Importance of Data Storytelling Components of a Data Story: Data, Narrative, and Visuals 11.1.2 Crafting a Narrative with Data Identifying the Main Message and Key Points Structuring the Story: Beginning, Middle, and End Techniques for Engaging the Audience 11.1.3 Using Visuals to Enhance the Narrative Choosing the Right Visualization for Your Data Best Practices for Designing Clear and Informative Visuals Integrating Visuals Seamlessly into the Story 11.1.4 Techniques for Simplifying Complex Data Avoiding Information Overload Highlighting Key Insights and Trends Using Analogies and Metaphors to Explain Data 11.1.5 Case Studies in Data Storytelling Examples of Effective Data Stories from Various Industries Analysis of What Makes These Stories Compelling 11.2 Creating Compelling Reports and Presentations 11.2.1 Principles of Report Writing Structure and Organization of Reports Clarity and Conciseness in Writing Incorporating Data Visualizations into Reports 11.2.2 Designing Effective Presentations Principles of Good Presentation Design Balancing Text, Images, and Data Visualizations Using Slide Transitions and Animations Effectively 11.2.3 Tools for Creating Reports and Presentations Overview of Tools: Microsoft PowerPoint, Google Slides, Tableau, Power BI		

Tips and Tricks for Using Each Tool Efficiently
Integrating Interactive Elements in Presentations
11.2.4 Tailoring Reports and Presentations to the Audience
Understanding the Needs and Interests of Your Audience
Customizing Content for Different Stakeholder Groups
Using Language and Examples that Resonate with the Audience
11.2.5 Reviewing and Refining Reports and Presentations
Techniques for Self-Review and Peer Review
Iterative Improvement: Incorporating Feedback
Ensuring Consistency and Professionalism
11.3 Presenting Findings to Stakeholders
11.3.1 Preparation for Stakeholder Presentations
Understanding Stakeholder Objectives and Concerns
Planning and Structuring the Presentation
Rehearsing and Refining Your Delivery
11.3.2 Techniques for Effective Oral Communication
Public Speaking Tips and Strategies
Using Body Language and Voice Modulation
Managing Nervousness and Building Confidence
11.3.3 Engaging Stakeholders During the Presentation
Techniques for Capturing and Maintaining Attention
Encouraging Questions and Interactive Discussion
Handling Difficult Questions and Feedback Gracefully
11.3.4 Follow-Up After the Presentation
Summarizing Key Takeaways and Next Steps
Providing Supplementary Materials and Detailed Reports
Gathering Feedback for Continuous Improvement
11.3.5 Case Studies of Successful Stakeholder Presentations
Analysis of Real-World Examples
Identifying Best Practices and Lessons Learned
12.1 Ethical Considerations in Data Analysis
12.2 Ensuring Data Privacy and Security
12.3 Best Practices for Sustainable BI Practices.

Books Recommended:

1. **"Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking"**
    - Authors: Foster Provost and Tom Fawcett
    - Publisher: O'Reilly Media
    - Description: This book provides a solid foundation in data science principles, focusing on data mining techniques and their application in business contexts. It is a practical guide for understanding data-analytic thinking.
2. **"Business Intelligence: A Managerial Approach"**
    - Authors: Efraim Turban, Ramesh Sharda, Dursun Delen, and David King
    - Publisher: Pearson
    - Description: This book covers the comprehensive architecture of business intelligence, including methodologies, tools, and applications. It is an excellent resource for understanding BI from a managerial perspective.
3. **"The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling"**
    - Authors: Ralph Kimball and Margy Ross
    - Publisher: Wiley
    - Description: A classic book on data warehousing and dimensional modeling, this text is essential for students looking to understand the structure and design of data warehouses, which are crucial for BI.
4. **"Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython"**
    - Author: Wes McKinney
    - Publisher: O'Reilly Media
    - Description: This book provides a thorough introduction to data analysis in Python, focusing on libraries like pandas and NumPy. It is ideal for hands-on learning and practical application of data analysis techniques.
5. **"Storytelling with Data: A Data Visualization Guide for Business Professionals"**
    - Author: Cole Nussbaumer Knaflic
    - Publisher: Wiley
    - Description: This book emphasizes the importance of data visualization and effective communication. It provides practical tips and strategies for creating compelling data stories.
6. **"Data Mining: Concepts and Techniques"**
    - Authors: Jiawei Han, Micheline Kamber, and Jian Pei
    - Publisher: Morgan Kaufmann
    - Description: This comprehensive guide to data mining covers the theoretical foundations and practical applications of various data mining techniques. It is suitable for both beginners and advanced learners.
7. **"Practical Business Intelligence"**
    - Authors: Ahmed Sherif, R. Kelly Rainer Jr., and Brad Prince
    - Publisher: McGraw Hill Education
    - Description: This book focuses on the practical aspects of implementing business intelligence solutions, including project management, data integration, and real-world case studies. It is designed for students and professionals looking to apply BI concepts in business settings.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2

5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 511PE: GRAPH THEORY (Professional Elective – I)
B.Tech. III Year I Sem.

Teaching Scheme:	**Credit**	**Examination Scheme:**
TH: - 4 Hours/Week	**TH: 04**	**In Sem. Evaluation: 25 Marks** **Mid Sem. Exam: 25 Marks** **End Sem. Exam: 50 Marks** **Total : 100 Marks**

Course Prerequisites: None

Course Objective:

- Understanding graphs, trees, connected paths, applications of trees and graphs.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Know some important classes of graph theoretic problems;
- Prove central theorems about trees, matching, connectivity, coloring and planar graphs;
- Describe and apply some basic algorithms for graphs;
- Use graph theory as a modeling tool.

Course Contents

UNIT-I	**Introduction**	**07 Hours**
Discovery of graphs, Definitions, Subgraphs, Isomorphic graphs, Matrix representations of graphs, Degree of a vertex, Directed walks, paths and cycles, Connectivity in digraphs, Eulerian and Hamilton digraphs, Eulerian digraphs, Hamilton digraphs, Special graphs, Complements, Larger graphs from smaller graphs, Union, Sum, Cartesian Product, Composition, Graphic sequences, Graph theoretic model of the LAN problem, Havel-Hakimi criterion, Realization of a graphic sequence.		
UNIT-II	**Connected graphs and shortest paths**	**08 Hours**
Walks, trails, paths, cycles, Connected graphs, Distance, Cut-vertices and cut-edges, Blocks, Connectivity, Weighted graphs and shortest paths, Weighted graphs, Dijkstra"s shortest path algorithm, Floyd-Warshall shortest path algorithm.		
UNIT-III	**Trees**	**08 Hours**

Definitions and characterizations, Number of trees, Cayley‟s formula, Kircho↲-matrix-tree theorem, Minimum spanning trees, Kruskal‟s algorithm, Prim‟s algorithm, Special classes of graphs, Bipartite Graphs, Line Graphs, Chordal Graphs, Eulerian Graphs, Fleury‟s algorithm, Chinese Postman problem, Hamilton Graphs, Introduction, Necessary conditions and sufficient conditions.		
UNIT-IV	**Independent sets coverings and matchings**	**07 Hours**
Introduction, Independent sets and coverings: basic equations, Matchings in bipartite graphs, Hall‟s Theorem, K¨onig‟s Theorem, Perfect matchings in graphs, Greedy and approximation algorithms.		
UNIT-V	**Vertex Colorings**	**08 Hours**
Basic definitions**,** Cliques and chromatic number, Mycielski‟s theorem, Greedy coloring algorithm, Coloring of chordal graphs, Brooks theorem, Edge Colorings, Introduction and Basics, Gupta-Vizing theorem, Class-1 and Class-2 graphs, Edge-coloring of bipartite graphs, Class-2 graphs, Hajos union and Class-2 graphs, A scheduling problem and equitable edge-coloring.		

TEXT BOOKS:

1. J. A. Bondy and U. S. R. Murty. Graph Theory, volume 244 of Graduate Texts in Mathematics. Springer, 1st edition, 2008.
2. J. A. Bondy and U. S. R. Murty. Graph Theory with Applications.

REFERENCE BOOKS:

1. Lecture Videos: http://nptel.ac.in/courses/111106050/13
2. Introduction to Graph Theory, Douglas B. West, Pearson.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 512PE: ADVANCED COMPUTER ARCHITECTURE
(Professional Elective – I)
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Computer Organization

Course Objective:

- Impart the concepts and principles of parallel and advanced computer architectures.
- Develop the design techniques of Scalable and multithreaded Architectures.
- Apply the concepts and techniques of parallel and advanced computer architectures to design modern computer systems.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Computational models and Computer Architectures.
- Concepts of parallel computer models.
- Scalable Architectures, Pipelining, Superscalar processors

Course Contents

UNIT-I	SIMD/ PRAM/ VLSI models	07 Hours

Theory of Parallelism, Parallel computer models, The State of Computing, Multiprocessors and Multicomputers, Multivector and SIMD Computers, PRAM and VLSI models, Architectural development tracks, Program and network properties, Conditions of parallelism, Program partitioning and Scheduling, Program flow Mechanisms, System interconnect Architectures.

UNIT-II	Principles of Scalable performance	08 Hours

Performance metrics and measures, Parallel Processing applications, Speed up performance laws, Scalability Analysis and Approaches, Hardware Technologies, Processes and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors.

UNIT-III	Shared-Memory Organizations	08 Hours

Sequential and weak consistency models, Pipelining and superscalar techniques, Linear Pipeline Processors, Non-Linear Pipeline Processors, Instruction Pipeline design, Arithmetic pipeline design, superscalar pipeline design.

UNIT-IV	Parallel and Scalable Architectures	07 Hours
Multiprocessors and Multicomputers, Multiprocessor system interconnects, cache coherence and synchronization mechanism, Three Generations of Multicomputers, Message-passing Mechanisms, Multivetor and SIMD computers.		
UNIT-V	**Vector Processing Principles**	**08 Hours**
Multivector Multiprocessors, Compound Vector processing, SIMD computer Organizations, The connection machine CM-5.		

TEXT BOOK:

1. Advanced Computer Architecture, Kai Hwang, 2nd Edition, Tata McGraw Hill Publishers.

REFERENCE BOOKS:

1. Computer Architecture, J.L. Hennessy and D.A. Patterson, 4th Edition, ELSEVIER.
2. Advanced Computer Architectures, S.G.Shiva, Special Indian edition, CRC, Taylor &Francis.
3. Introduction to High Performance Computing for Scientists and Engineers, G. Hager and G. Wellein, CRC Press, Taylor & Francis Group.
4. Advanced Computer Architecture, D. Sima, T. Fountain, P. Kacsuk, Pearson education.
5. Computer Architecture, B. Parhami, Oxford Univ. Press.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 513PE: WEB PROGRAMMING (Professional Elective – I)
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Basic Web Programming

Course Objective:

- Understand the technologies used in Web Programming.
- Know the importance of object-oriented aspects of Scripting.
- Understand creating database connectivity using JDBC.
- Learn the concepts of web-based application using sockets.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Design web pages.
- Use technologies of Web Programming.
- Apply object-oriented aspects to Scripting.
- Create databases with connectivity using JDBC.
- Build web-based application using sockets.

Course Contents

UNIT-I	Client-side Programming	07 Hours

HTML- Basic Tags- List, Tables, Images, Forms, Frames, CSS
JAVA Script -
Web page Designing using HTML, Scripting basics- Client side and server side scripting. Java ScriptObject, names, literals, operators and expressions- statements and features- events - windows - documents - frames - data types - built-in functions- Browser object model - Verifying forms.-HTML5- CSS3- HTML 5 canvas - Web site creation using tools.

UNIT-II	JAVA	08 Hours

Introduction to object-oriented programming-Features of Java – Data types, variables and arrays – Operators – Control statements – Classes and Methods – Inheritance. Packages and Interfaces – Exception Handling – Multithreaded Programming – Input/Output – Files – Utility Classes – String Handling.

UNIT-III	JDBC	08 Hours

JDBC Overview – JDBC implementation – Connection class – Statements - Catching Database Results, handling database Queries. Networking– InetAddress class – URL class- TCP sockets – UDP sockets, Java Beans –RMI.		
UNIT-IV	**APPLETS**	**07 Hours**
Java applets- Life cycle of an applet — Adding images to an applet — Adding sound to an applet. Passing parameters to an applet. Event Handling. Introducing AWT: Working with Windows Graphics and Text. Using AWT Controls, Layout Managers and Menus. Servlet — life cycle of a servlet. The Servlet API, Handling HTTP Request and Response, using Cookies, Session Tracking. Introduction to JSP.		
UNIT-V	**XML AND WEB SERVICES**	**08 Hours**
Xml – Introduction-Form Navigation-XML Documents- XSL – XSLT- Web services-UDDI-WSDL-Java web services — Web resources.		

TEXT BOOKS:

1. Harvey Deitel, Abbey Deitel, Internet and World Wide Web: How To Program 5th Edition.
2. Herbert Schildt, Java - The Complete Reference, 7th Edition. Tata McGraw- Hill Edition.
3. Michael Morrison XML Unleashed Tech media SAMS.

REFERENCE BOOKS:

1. John Pollock, Javascript - A Beginners Guide, 3rd Edition –- Tata McGraw-Hill Edition.
2. Keyur Shah, Gateway to Java Programmer Sun Certification, Tata McGraw Hill, 2002.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students’ reading and writing

skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 514PE: IMAGE PROCESSING (Professional Elective – I)
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

- **Course Prerequisites:** Students are expected to have knowledge in linear signals and systems, Fourier Transform, basic linear algebra, basic probability theory and basic programming techniques; knowledge of digital signal processing is desirable.
- A course on "Computational Mathematics"
- A course on "Computer Oriented Statistical Methods"

Course Objective:

- Provide a theoretical and mathematical foundation of fundamental Digital Image Processing concepts.
- The topics include image acquisition; sampling and quantization; preprocessing; enhancement; restoration; segmentation; and compression.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Demonstrate the knowledge of the basic concepts of two-dimensional signal acquisition, sampling, and quantization.
- Demonstrate the knowledge of filtering techniques.
- Demonstrate the knowledge of 2D transformation techniques.
- Demonstrate the knowledge of image enhancement, segmentation, restoration and compression techniques.

Course Contents

UNIT-I	Digital Image Fundamentals	07 Hours

Digital Image through Scanner, Digital Camera. Concept of Gray Levels.Gray Level to Binary Image Conversion. Sampling and Quantization. Relationship between Pixels.Imaging Geometry. 2D Transformations-DFT, DCT, KLT and SVD.

UNIT-II	Image Processing	08 Hours

Image Enhancement in Spatial Domain Point Processing, Histogram Processing, Spatial Filtering, Enhancement in Frequency Domain, Image Smoothing, Image Sharpening.

UNIT-III	Image Restoration Degradation Model	08 Hours

Algebraic Approach to Restoration, Inverse Filtering, Least Mean Square Filters, Constrained Least Squares Restoration, Interactive Restoration.		
UNIT-IV	**Image Segmentation Detection of Discontinuities**	**07 Hours**
Edge Linking and Boundary Detection, Thresholding, Region Oriented Segmentation.		
UNIT-V	**Image Compression**	**08 Hours**
Image Compression Redundancies and their Removal Methods, Fidelity Criteria, Image Compression Models, Source Encoder and Decoder, Error Free Compression, Lossy Compression.		

TEXT BOOK:

1. Digital Image Processing: R.C. Gonzalez & R. E. Woods, Addison Wesley/ Pearson Education,2nd Ed, 2004.

REFERENCE BOOKS:

1. Fundamentals of Digital Image Processing: A. K. Jain, PHI.
2. Digital Image Processing using MAT LAB: Rafael C. Gonzalez, Richard E. Woods, Steven L.Eddins: Pearson Education India, 2004.
3. Digital Image Processing: William K. Pratt, John Wilely, 3rd Edition, 2004.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO’s-PO’s & PSO’s MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 515PE: COMPUTER GRAPHICS (Professional Elective – I)

B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Programming for problem solving and Data Structures

Course Objective:

- Provide the basics of graphics systems including Points and lines, line drawing algorithms, 2D, 3D objective transformations

Learning Course Outcome:
After successful completion of the course, students will able to:

- Explore applications of computer graphics
- Understand 2D, 3D geometric transformations and clipping algorithms
- Understand 3D object representations, curves, surfaces, polygon rendering methods, color models
- Analyze animation sequence and visible surface detection methods

Course Contents

UNIT-I	Introduction	07 Hours

Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random-scan systems, graphics monitors and work stations and input devices

Output primitives: Points and lines, line drawing algorithms (DDA and Bresenham's Algorithm) circle- generating algorithms and ellipse - generating algorithms
Polygon Filling: Scan-line algorithm, boundary-fill and flood-fill algorithms

UNIT-II	transformations	08 Hours

2-D geometric transformations: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems

2-D viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, clipping operations, point clipping, Line clipping-Cohen Sutherland algorithms, Polygon clipping-Sutherland Hodgeman polygon clipping algorithm.

UNIT-III	object representation	08 Hours

3-D object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces, Polygon rendering methods, color models and color applications.

UNIT-IV	Geometric transformations	07 Hours

3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations.
3-D viewing: Viewing pipeline, viewing coordinates, projections, view volume and general projection transforms and clipping.

UNIT-V	Computer animation	08 Hours

Design of animation sequence, general computer animation functions, raster animations, computer animation languages, key frame systems, motion specifications.
Visible surface detection methods: Classification, back-face detection, depth-buffer method, BSP- tree method, area sub-division method and octree method.

TEXT BOOKS:

1. "Computer Graphics C version", Donald Hearn and M. Pauline Baker, Pearson Education

REFERENCE BOOKS:

1. Procedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2nd edition.
2. Principles of Interactive Computer Graphics", Neuman and Sproul, TMH.
3. Principles of Computer Graphics, Shalini Govil, Pai, 2005, Springer.
4. "Computer Graphics Principles & practice", second edition in C, Foley, Van Dam, Feiner and Hughes, Pearson Education.
5. Computer Graphics, Steven Harrington, TMH.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing

skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 502L: R PROGRAMMING LAB
B.Tech. III Year I Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Any programming language.

Course Objective:

- Familiarize with R basic programming concepts, various data structures for handling datasets, various graph representations and Exploratory Data Analysis concepts.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Setup R programming environment.
- Understand and use R – Data types and R – Data Structures.
- Develop programming logic using R – Packages.
- Analyze data sets using R – programming capabilities

Course Contents

LIST OF EXPERIMENTS	07 Hours

1. Download and install R-Programming environment and install basic packages using install.packages() command in R.

2. Learn all the basics of R-Programming (Data types, Variables, Operators etc,.)

3. Write R command to
i) Illustrate summation, subtraction, multiplication, and division operations on vectors using vectors.
ii) Enumerate multiplication and division operations between matrices and vectors in R console

4. Write R command to
i) Illustrates the usage of Vector subsetting and Matrix subsetting
ii) Write a program to create an array of 3×3 matrices with 3 rows and 3 columns.

5. Write an R program to draw i) Pie chart ii) 3D Pie Chart, iii) Bar Chart along with chart legend by considering suitable CSV file

6. Create a CSV file having Speed and Distance attributes with 1000 records. Write R program to draw
i) Box plots

ii) Histogram
iii) Line Graph
iv) Multiple line graphs
v) Scatter plot
to demonstrate the relation between the cars speed and the distance.

7. Implement different data structures in R (Vectors, Lists, Data Frames)

8. Write an R program to read a csv file and analyze the data in the file using EDA (Explorative Data Analysis) techniques.

9. Write an R program to illustrate Linear Regression and Multi linear Regression considering suitable CSV file.

TEXT BOOKS:

1. R Programming for Data Science by Roger D. Peng
2. The Art of R Programming by Norman Matloff Cengage Learning India.

REFERENCE BOOKS:

1. Hadley Wickham, Garrett Grolemund, R for Data Science: Import, Tidy, Transform, Visualize, and Model Data 1st Edition, O'Reilly
2. Tilman M. Davies, The book of R a first course in programming and statistics, no starch press

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 503L: DESCRIPTIVE ANALYTICS LAB
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Computer Networks Lab Practice

Course Objective:

- To understand the working principle of various communication protocols.
- To understand the network simulator environment and visualize a network topology and observe its performance
- To analyze the traffic flow and the contents of protocol frames

Learning Course Outcome:

After successful completion of the course, students will able to:

- Implement data link layer farming methods
- Analyze error detection and error correction codes.
- Implement and analyze routing and congestion issues in network design.
- Implement Encoding and Decoding techniques used in presentation layer
- To be able to work with different network tools

Course Contents

List of Experiments	07 Hours

Lab 1: Introduction to R and Python

Objective: Familiarize students with basic operations and functions in R and Python.

Tools: R, Python

1. Install and set up R and Python.
2. Write simple scripts to perform basic arithmetic operations.
3. Load and explore datasets using R and Python.

Lab 2: Data Collection and Importing Data

Objective: Learn methods to collect and import data from various sources.

Tools: R, Python, Excel

1. Import data from CSV, Excel, and web sources.
2. Clean and preprocess the imported data.
3. Handle missing values and remove duplicates.

Lab 3: Data Cleaning and Preparation

Objective: Perform data cleaning and preparation tasks.

Tools: R, Python

1. Identify and handle missing values.
2. Detect and remove outliers.
3. Standardize and normalize data.

Lab 4: Exploratory Data Analysis (EDA)

Objective: Conduct exploratory data analysis to understand data patterns.
Tools: R, Python

1. Compute summary statistics (mean, median, mode).
2. Create histograms, box plots, and scatter plots.
3. Identify patterns and correlations in the data.

Lab 5: Descriptive Statistics
Objective: Apply descriptive statistical methods to summarize data.
Tools: R, Python

1. Calculate measures of central tendency (mean, median, mode).
2. Calculate measures of dispersion (range, variance, standard deviation).
3. Visualize the results using appropriate charts.

Lab 6: Probability Distributions
Objective: Understand and apply different probability distributions.
Tools: R, Python

1. Generate and plot normal, binomial, and Poisson distributions.
2. Fit data to these distributions.
3. Analyze real-world data using these distributions.

Lab 7: Hypothesis Testing
Objective: Conduct hypothesis testing on datasets.
Tools: R, Python

1. Perform t-tests and chi-square tests.
2. Interpret p-values and confidence intervals.
3. Apply these tests to real-world data.

Lab 8: Data Visualization with Excel
Objective: Create basic data visualizations using Excel.
Tools: Excel

1. Create bar charts, line charts, and pie charts.
2. Use pivot tables to summarize data.
3. Customize charts for better readability.

Lab 9: Advanced Data Visualization with Python
Objective: Use Python libraries for advanced data visualization.
Tools: Python (matplotlib, seaborn)

1. Create complex plots like pair plots and heatmaps.
2. Customize visualizations with colors and themes.
3. Use subplots to compare multiple plots.

Lab 10: Data Visualization with Tableau
Objective: Create interactive data visualizations using Tableau.
Tools: Tableau

1. Import data into Tableau.
2. Create dashboards and stories.
3. Add interactivity to visualizations with filters and actions.

Lab 11: Data Visualization with Power BI
Objective: Develop interactive reports and dashboards using Power BI.
Tools: Power BI

1. Import and transform data in Power BI.

2. Create reports using different visualizations.
3. Develop interactive dashboards with slicers and filters.

Lab 12: Clustering Analysis
Objective: Perform clustering analysis to group similar data points.
Tools: R, Python (scikit-learn)
1. Implement K-means clustering.
2. Visualize the clusters.
3. Evaluate clustering results using silhouette score.

Lab 13: Classification Analysis
Objective: Apply classification techniques to labeled data.
Tools: R, Python (scikit-learn)
1. Implement decision trees and logistic regression.
2. Evaluate models using confusion matrix and accuracy.
3. Visualize decision boundaries.

Lab 14: Association Rule Mining
Objective: Discover associations and relationships in large datasets.
Tools: R, Python
1. Apply the Apriori algorithm.
2. Extract frequent itemsets and association rules.
3. Visualize association rules.

Lab 15: Anomaly Detection
Objective: Identify outliers and anomalies in data.
Tools: R, Python
1. Implement isolation forest and DBSCAN algorithms.
2. Visualize anomalies in the dataset.
3. Analyze and interpret the results.

Lab 16: Creating Reports and Dashboards in Tableau
Objective: Design professional reports and dashboards.
Tools: Tableau
1. Create comprehensive reports with multiple visualizations.
2. Develop interactive dashboards.
3. Share reports and dashboards with stakeholders.

Lab 17: Creating Reports and Dashboards in Power BI
Objective: Design and implement interactive reports and dashboards.
Tools: Power BI
1. Develop detailed reports with various visualizations.
2. Implement interactive elements like slicers.
3. Publish and share dashboards with stakeholders.

Lab 18: Data Storytelling with Visualizations
Objective: Learn techniques for effective data storytelling.
Tools: Tableau, Power BI
1. Create a narrative using visualizations.
2. Develop a cohesive story through a series of visualizations.
3. Present the story to a simulated audience.

Lab 19: Real-World Data Analysis Project 1

Objective: Apply learned skills to analyze a real-world dataset.
Tools: R, Python, Tableau, Power BI
1. Collect and clean a real-world dataset.
2. Perform exploratory data analysis.
3. Present findings through visualizations and reports.

Lab 20: Real-World Data Analysis Project 2
Objective: Conduct a comprehensive analysis on a different dataset.
Tools: R, Python, Tableau, Power BI
1. Prepare and preprocess the dataset.
2. Apply statistical and data mining techniques.
3. Develop a dashboard to present insights.

Lab 21: Advanced Data Mining Techniques
Objective: Use advanced data mining methods to uncover insights.
Tools: R, Python
1. Implement clustering, classification, and regression.
2. Analyze complex datasets.
3. Interpret the results and derive insights.

Lab 22: BI Project Management
Objective: Manage and execute a BI project from start to finish.
Tools: Excel, Power BI
1. Plan a BI project.
2. Develop and implement the project.
3. Measure the project's success and value.

Lab 23: Evaluating BI Success and Value
Objective: Measure the effectiveness of BI initiatives.
Tools: Excel, Power BI
1. Define metrics for success.
2. Collect and analyze data related to BI initiatives.
3. Present evaluation results.

TEXT BOOK:
1. Computer Networks, Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI.

REFERENCE BOOKS:
1. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.
2. Data Communications and Networking – Behrouz A. Forouzan. 3rd Edition, TMH.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2

Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 505L: ETL- KAFKA/TALEND
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Develop a comprehensive understanding of Extract, Transform, Load (ETL) processes using Apache Kafka and Talend.
- Understand how to scale Kafka clusters seamlessly to handle growing data volumes, ensuring optimal performance for ETL operations.

Learning Course Outcome:
After successful completion of the course, students will able to:

1. Learn to design and deploy fault-tolerant Kafka clusters, ensuring data integrity and availability in real-world scenarios.
2. Gain practical experience in cluster management, topic creation, and basic operations such as producing and consuming messages.

Course Contents

LIST OF EXPERIMENTS	07 Hours

1. Install Apache Kafka on a single node.
2. Demonstrate setting up a single-node, single-broker Kafka cluster and show basic operations such as creating topics and producing/consuming messages.
3. Extend the cluster to multiple brokers on a single node.
4. Write a simple Java program to create a Kafka producer and Produce messages to a topic.
5. Implement sending messages both synchronously and asynchronously in the producer.
6. Develop a Java program to create a Kafka consumer and subscribe to a topic and consume messages.
7. Write a script to create a topic with specific partition and replication factor settings.
8. Simulate fault tolerance by shutting down one broker and observing the cluster behavior.
9. Implement operations such as listing topics, modifying configurations, and deleting topics.
10. Introduce Kafka Connect and demonstrate how to use connectors to integrate with external systems.
11. Implement a simple word count stream processing application using Kafka Stream
12. Implement Kafka integration with the Hadoop ecosystem.

TEXT BOOK:

1. Neha Narkhede, Gwen Shapira, Todd Palino, Kafka – The Definitive Guide: Real-time data and stream processing at scale, O′Reilly

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 601: AUTOMATA THEORY AND COMPILER DESIGN
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- To introduce the fundamental concepts of formal languages, grammars and automata theory.
- To understand deterministic and non-deterministic machines and the differences between decidability and undecidability.
- Introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler.
- Topics include phases of compiler, parsing, syntax directed translation, type checking use of symbol tables, intermediate code generation

Learning Course Outcome:
After successful completion of the course, students will able to:

- Able to employ finite state machines for modeling and solving computing problems.
- Able to design context free grammars for formal languages.
- Able to distinguish between decidability and undecidability.
- Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis.
- Acquire skills in using lex tool and design LR parsers

Course Contents

UNIT-I	Finite Automata	07 Hours

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.
Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.
Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with €-transitions to NFA without €-transitions. Conversion of NFA to DFA

UNIT-II	Regular Expressions	08 Hours

Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions. **Pumping Lemma for Regular Languages**: Statement of the pumping lemma, Applications of the Pumping Lemma. **Context-Free Grammars:** Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.		
UNIT-III	**Automata**	**08 Hours**
Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state **Turing Machines**: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine **Undecidability:** Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines		
UNIT-IV	**Introduction: The structure of a compiler**	**07 Hours**
Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex **Syntax Analysis:** Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers		
UNIT-V	**Syntax-Directed Translation**	**08 Hours**
Syntax-Directed Definitions, Evaluation Orders for SDD's, Syntax- Directed Translation Schemes, Implementing L-Attributed SDD's. **Intermediate-Code Generation:** Variants of Syntax Trees, Three-Address Code **Run-Time Environments:** Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management.		

TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation, 3rd. Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Theory of Computer Science- Automata languages and computation, Mishra and Chandrashekaran, 2nd Edition, PHI.

REFERENCE BOOKS:

1. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, 2nd Edition, Pearson.
2. Introduction to Formal languages Automata Theory and Computation, Kamala Krithivasan, Rama R, Pearson.
3. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
4. lex & yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
5. Compiler Construction, Kenneth C. Louden, Thomson. Course Technology.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 602: MACHINE LEARNING
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: 10 + 2 Statistics

Course Objective:

- Understand the core principles and theories behind machine learning algorithms.
- Gain practical experience in implementing and applying supervised and unsupervised learning techniques.
- Explore advanced topics such as deep learning and reinforcement learning.
- Learn how to evaluate and optimize machine learning models for performance and accuracy.
- Develop critical thinking and problem-solving skills in the context of data-driven decision making.
- Explore ethical considerations and societal implications of machine learning technologies.
- Build a foundation for further study or career advancement in the field of machine learning and artificial intelligence.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Implement various machine learning algorithms for regression, classification, and clustering tasks.
- Evaluate and compare the performance of different machine learning models using appropriate metrics.
- Apply deep learning techniques to solve complex problems in image recognition, natural language processing, and other domains.
- Understand the principles of reinforcement learning and its applications in areas such as gaming and robotics.
- Interpret and communicate the results of machine learning experiments effectively.
- Demonstrate awareness of ethical considerations and biases inherent in machine learning systems.
- Apply machine learning techniques to real-world datasets to extract meaningful insights and make data-driven decisions.

Course Contents

UNIT-I	Introduction to Machine Learning	07 Hours

- **Introduction and Motivation**
  - Definition of Machine Learning
  - Motivation for Machine Learning
  - Historical Context of Machine Learning Techniques
- **Applications of Machine Learning**
  - Broad Application Areas
  - Specific Use Cases
  - Ethical Considerations and Social Implications
- **The Origin and Evolution of Machine Learning**
  - Early Foundations
  - Developments in Statistical Learning Theory

- Rise of Neural Networks
- Contemporary Trends and Future Directions
- **Foundational Concepts**
  - Abstraction in Machine Learning
  - Knowledge Representation
  - Generalization in Machine Learning
  - Model Complexity and Interpretability
- **Assessing Learning Success**
  - Evaluation Metrics and Performance Measures
  - Cross-Validation Techniques
  - Model Selection and Hyperparameter Tuning
  - Interpretability and Explainability
- **Steps to Apply Machine Learning**
  - Data Preprocessing and Cleaning
  - Model Training and Evaluation Pipeline
  - Deployment and Monitoring
- **Overview of Machine Learning Methods**
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning

UNIT-II	**Supervised Learning**	**08 Hours**

- **Introduction to Supervised Learning**
  - Overview of Supervised Learning
  - Components of Supervised Learning
  - Importance and Applications
- **Regression**
  - Overview of Regression Analysis
  - Simple Linear Regression
  - Multiple Regression
  - Nonlinear Regression
- **Classification**
  - Overview of Classification
  - Decision Trees
  - Support Vector Machines (SVMs)
  - Logistic Regression
  - Performance Evaluation
- **Advanced Techniques in Classification**
  - Ensemble Learning Methods
  - Neural Networks for Classification

UNIT-III	**Unsupervised Learning**	**08 Hours**

- **Introduction to Unsupervised Learning**
  - Overview of Unsupervised Learning
  - Importance and Applications
- **Clustering Techniques**
  - Overview of Clustering
  - Major Clustering Approaches
  - Evaluating Cluster Quality
- **Dimensionality Reduction**
  - Overview of Dimensionality Reduction
  - Linear Techniques
  - Nonlinear Techniques
- **Advanced Clustering Methods**
  - Soft Clustering vs. Hard Clustering
  - Fuzzy C-Means Clustering
  - Gaussian Mixture Models (GMM)
  - Spectral Clustering

UNIT-IV	Advanced Topics in Machine Learning	07 Hours

- **Ensemble Learning Methods**
  - Introduction to Ensemble Learning
  - Bagging and Boosting
- **Deep Learning**
  - Introduction to Deep Learning
  - Neural Networks, CNNs, RNNs
- **Reinforcement Learning**
  - Introduction to Reinforcement Learning
  - Q-Learning, DQNs, Policy Gradient Methods
- **Advanced Techniques in Model Interpretability**
  - Model-Agnostic Interpretability Techniques
  - Model-Specific Interpretability Techniques

UNIT-V	Model Evaluation and Selection	08 Hours

- **Model Evaluation Metrics**
  - Introduction to Evaluation Metrics
  - Classification Evaluation Metrics
  - Regression Evaluation Metrics
- **Cluster Validation Measures**
  - Introduction to Cluster Validation
  - External and Internal Cluster Validation Measures
  - Relative Validity Measures
- **Regression Model Assessment Techniques**
  - Overview of Regression Model Assessment
  - Hypothesis Testing in Regression Models

- Model Selection Techniques
- Cross-Validation Techniques

- **Hyperparameter Tuning and Optimization**
  - Introduction to Hyperparameter Tuning
  - Grid Search and Random Search
  - Bayesian Optimization
  - Automated Hyperparameter Tuning Frameworks

UNIT-VI	Advanced Concepts and Applications	08 Hours

- **Transfer Learning and Domain Adaptation**
  - Introduction to Transfer Learning
  - Techniques in Transfer Learning
  - Applications of Transfer Learning
- **Meta-Learning and Learning to Learn**
  - Overview of Meta-Learning
  - Techniques in Meta-Learning
  - Applications of Meta-Learning
- **Adversarial Machine Learning and Robustness**
  - Understanding Adversarial Machine Learning
  - Defenses Against Adversarial Attacks
  - Applications and Implications
- **Fairness, Accountability, and Transparency**
  - Fairness in Machine Learning
  - Accountability in Machine Learning
  - Ethical Considerations and Social Implications

TEXT BOOKS:

1. Stephen Marsland, —Machine Learning — An Algorithmic Perspective, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.

REFERENCE BOOKS:

2. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron
3. "Pattern Recognition and Machine Learning" by Christopher M. Bishop
4. "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville
5. "Python Machine Learning" by Sebastian Raschka and Vahid Mirjalili
6. "Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy
7. "Introduction to Machine Learning with Python: A Guide for Data Scientists" by Andreas C. Müller and Sarah Guido
8. "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 603: PREDICTIVE ANALYTICS
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Understand the fundamental concepts and importance of predictive analytics in various industries.
- Learn methods for data collection, cleaning, and preprocessing, including handling missing values and normalization.
- Gain proficiency in exploratory data analysis and feature engineering.
- Master the application of probability theory and statistical methods in predictive modeling.
- Develop skills in building and interpreting regression models, including simple and multiple linear regression.
- Explore machine learning algorithms for supervised and unsupervised learning, as well as ensemble methods.
- Understand and apply time series analysis techniques for forecasting.
- Learn model evaluation and validation techniques to ensure robust predictive performance.
- Gain practical experience in deploying, monitoring, and updating predictive models.
- Apply predictive analytics techniques to solve real-world problems in various industries, such as business analytics, financial forecasting, marketing, healthcare, and e-commerce.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Explain the significance and applications of predictive analytics in decision-making processes.
- Prepare and preprocess datasets for predictive modeling tasks.
- Apply various data mining and machine learning techniques to extract patterns and build predictive models.
- Evaluate and compare predictive models using appropriate metrics and validation techniques.
- Develop and interpret predictive models using regression analysis and advanced machine learning algorithms.
- Implement time series forecasting models and understand their applications.
- Deploy predictive models and monitor their performance in a real-world setting.
- Apply predictive analytics techniques to specific industry problems and present findings effectively.

Course Contents

UNIT-I	Introduction to Predictive Analytics, Data Understanding and Preparation	07 Hours

1.1 Overview of Predictive Analytics
Definition of Predictive Analytics: Understanding what predictive analytics is and its role in data analysis.
History and Evolution: A brief history of predictive analytics and its evolution over time.
Key Components: Introduction to the major components of predictive analytics, including data collection, data processing, model building, and model deployment.

1.2 Importance and Applications in Various Industries

Business:

Customer Retention: Using predictive analytics to predict customer churn and implement retention strategies.

Sales Forecasting: Forecasting future sales based on historical data.

Finance:

Risk Management: Predicting financial risks and defaults.

Fraud Detection: Identifying potential fraudulent activities.

Marketing:

Targeted Marketing: Personalizing marketing campaigns based on customer behavior predictions.

Market Basket Analysis: Understanding product purchase patterns.

Healthcare:

Predictive Diagnosis: Predicting disease outbreaks and patient health outcomes.

Resource Allocation: Optimizing the allocation of healthcare resources.

Others:

Manufacturing: Predicting equipment failures and maintenance needs.

Retail: Inventory management and demand forecasting.

Government: Predictive policing and public policy formulation.

1.3 Key Concepts and Terminology

Data Mining: Techniques for extracting patterns from large datasets.

Machine Learning: Algorithms that allow computers to learn from and make predictions based on data.

Big Data: Large and complex data sets that traditional data-processing software can't handle efficiently.

Predictive Model: A model that predicts future outcomes based on historical data.

Training Data: The data used to train predictive models.

Test Data: Data used to test the accuracy of predictive models.

Overfitting: A model that fits the training data too well but performs poorly on unseen data.

Underfitting: A model that is too simple to capture the underlying patterns in the data.

Feature Engineering: The process of selecting, modifying, or creating features (variables) for use in model building.

Cross-Validation: A technique for assessing how the results of a predictive model will generalize to an independent data set.

1.4 Steps in the Predictive Analytics Process

Problem Definition:

Understanding the Business Problem: Clearly defining the problem you aim to solve with predictive analytics.

Setting Objectives: Establishing what you want to achieve through predictive modeling.

Data Collection:

Sources of Data: Identifying and gathering relevant data from various sources.

Data Quality: Ensuring the accuracy, completeness, and reliability of the data collected.

Data Preparation:

Data Cleaning: Removing noise and correcting inconsistencies in the data.

Data Transformation: Normalizing and transforming data to a suitable format for analysis.

Feature Engineering: Creating new features from existing data to improve model performance.

Model Building:

Selecting Algorithms: Choosing appropriate algorithms based on the problem and data.

Training the Model: Using training data to create the predictive model.

Tuning Hyperparameters: Adjusting the model parameters to optimize performance.

Model Evaluation:
Validation Techniques: Using cross-validation and other techniques to evaluate model performance.
Performance Metrics: Assessing model accuracy, precision, recall, F1-score, ROC-AUC, etc.
Model Deployment:
Integration: Integrating the model into business processes or systems.
Monitoring: Continuously monitoring model performance and updating as needed.
Interpretation and Communication:
Result Interpretation: Understanding and interpreting the results generated by the model.
Reporting: Communicating findings and insights to stakeholders through reports and visualizations.

2.1 Data Collection Methods
Primary Data Collection:
Surveys and Questionnaires: Designing and conducting surveys for data collection.
Experiments: Collecting data through controlled experiments.
Observations: Gathering data through direct observation.
Secondary Data Collection:
Public Databases: Using data from public repositories and government databases.
Corporate Databases: Accessing internal corporate databases for historical data.
Web Scraping: Extracting data from websites and online sources.
Automated Data Collection:
Sensors and IoT Devices: Collecting data from interconnected devices.
APIs: Using Application Programming Interfaces for data retrieval from different platforms.
2.2 Data Cleaning and Preprocessing
Data Cleaning:
Handling Inconsistent Data: Correcting errors and inconsistencies in the data.
Removing Duplicates: Identifying and removing duplicate records.
Outlier Detection and Treatment: Detecting outliers and deciding whether to remove or transform them.
Data Transformation and Normalization:
Standardization: Transforming data to have a mean of zero and a standard deviation of one.
Min-Max Normalization: Scaling data to fit within a specified range, usually 0 to 1.
Log Transformation: Applying logarithmic transformation to stabilize variance and make the data more normal distribution-like.
Categorical Data Encoding: Converting categorical variables into numerical format using techniques like one-hot encoding and label encoding.
Handling Missing Values:
Identifying Missing Data: Detecting missing values in the dataset.
Imputation Techniques:
Mean/Median/Mode Imputation: Replacing missing values with the mean, median, or mode of the column.
K-Nearest Neighbors (KNN) Imputation: Using KNN algorithm to estimate and replace missing values.
Regression Imputation: Using regression models to predict and fill in missing values.
Multiple Imputation: Generating multiple imputations to account for the uncertainty of missing data.
2.3 Exploratory Data Analysis (EDA)
Descriptive Statistics:
Measures of Central Tendency: Calculating mean, median, and mode.
Measures of Dispersion: Understanding range, variance, and standard deviation.

Distribution Analysis: Analyzing the distribution of data using histograms, probability plots, and skewness/kurtosis measures.

Data Visualization Techniques:

Univariate Analysis: Visualizing single variable distributions using histograms, box plots, and density plots.

Bivariate Analysis: Exploring relationships between two variables using scatter plots, correlation matrices, and bar charts.

Multivariate Analysis: Visualizing relationships among multiple variables using pair plots, heatmaps, and parallel coordinates plots.

Time Series Visualization: Plotting time series data to identify trends and patterns over time.

2.4 Feature Engineering

Feature Selection:

Filter Methods: Using statistical tests and metrics (e.g., Chi-square test, ANOVA, correlation coefficient) to select relevant features.

Wrapper Methods: Employing algorithms like Recursive Feature Elimination (RFE) to identify the best subset of features.

Embedded Methods: Using algorithms that perform feature selection during the model training process (e.g., Lasso regression, decision trees).

Feature Extraction:

Principal Component Analysis (PCA): Reducing dimensionality by transforming features into a new set of orthogonal components.

Linear Discriminant Analysis (LDA): Reducing dimensionality while preserving class separability.

t-Distributed Stochastic Neighbor Embedding (t-SNE): Visualizing high-dimensional data in a lower-dimensional space.

Text Feature Extraction: Converting text data into numerical features using techniques like TF-IDF, word embeddings, and n-grams.

Image Feature Extraction: Using convolutional neural networks (CNNs) and other techniques to extract features from image data.

UNIT-II	Probability Theory and Statistics for Predictive Modeling, Regression Analysis, Probability Theory and Statistics for Predictive Modeling	08 Hours

3.1 Basics of Probability Theory

Foundational Concepts:

Probability Definitions: Understanding probability as a measure of uncertainty.

Sample Space: Defining the set of all possible outcomes.

Events: Identifying and describing events within a sample space.

Probability Rules:

Addition Rule: Calculating the probability of the union of two events.

Multiplication Rule: Calculating the probability of the intersection of two events.

Complementary Rule: Understanding the probability of the complement of an event.

Conditional Probability and Independence:

Conditional Probability: Calculating the probability of an event given another event has occurred.

Independence: Identifying independent events and understanding their implications.

3.2 Probability Distributions

Discrete and Continuous Distributions:

Discrete Distributions:

Binomial Distribution: Characteristics, mean, and variance.
Poisson Distribution: Characteristics, mean, and variance.
Continuous Distributions:
Uniform Distribution: Understanding the properties and applications.
Exponential Distribution: Characteristics and applications in time-between-events modeling.
Normal Distribution:
Properties of Normal Distribution: Mean, variance, skewness, and kurtosis.
Standard Normal Distribution: Z-scores and the standard normal table.
Central Limit Theorem: Understanding its significance in the context of predictive modeling.
Applications of Normal Distribution: Usage in confidence intervals, hypothesis testing, and error analysis.
3.3 Statistical Inference
Point Estimation and Confidence Intervals:
Point Estimation:
Estimators and Estimates: Understanding the concepts and properties of estimators.
Methods of Estimation: Maximum likelihood estimation (MLE) and method of moments.
Confidence Intervals:
Constructing Confidence Intervals: For population mean and proportion.
Interpretation: Understanding the confidence level and interval bounds.
Margin of Error: Calculating and interpreting the margin of error.
Hypothesis Testing:
Formulating Hypotheses:
Null Hypothesis (H0): Definition and formulation.
Alternative Hypothesis (H1): Definition and formulation.
Types of Errors:
Type I Error (α): Probability and implications.
Type II Error (β): Probability and implications.
Test Statistics and P-values:
Calculating Test Statistics: Using Z-tests, T-tests, Chi-square tests, and F-tests.
P-values: Interpretation and decision-making.
Decision Rules:
Significance Level: Understanding α and its role in hypothesis testing.
Critical Values: Using critical values to make decisions.
Power of the Test: Understanding and calculating the power of a test.

4.1 Simple Linear Regression
4.1.1 Model Assumptions
Linearity: Relationship between independent and dependent variables is linear.
Independence: Observations are independent of each other.
Homoscedasticity: Constant variance of errors.
Normality: Errors are normally distributed.
No Multicollinearity: For single predictor, this is not applicable.
4.1.2 Estimation of Parameters
Least Squares Method: Derivation and computation.
Interpretation of Coefficients: Slope and intercept.
Standard Error of Estimate: Calculation and interpretation.
Confidence Intervals for Parameters: Construction and interpretation.

4.1.3 Goodness of Fit: R-squared
Definition and Calculation: Proportion of variance explained by the model.
Adjusted R-squared: Adjusting for the number of predictors.
Interpretation: Assessing model fit.
4.2 Multiple Linear Regression
4.2.1 Model Building
Selecting Predictors: Criteria for inclusion.
Model Specification: Including interaction terms and polynomial terms.
Assumptions: Linearity, independence, homoscedasticity, normality, and no multicollinearity.
4.2.2 Interpretation of Coefficients
Partial Slopes: Meaning and interpretation in the context of other variables.
Standardized Coefficients: Comparison across variables.
Significance Testing: T-tests for individual coefficients.
4.2.3 Diagnostics and Residual Analysis
Residual Plots: Checking assumptions.
Influential Points: Leverage and Cook's distance.
Variance Inflation Factor (VIF): Detecting multicollinearity.
Normal Probability Plots: Checking normality of residuals.
4.3 Advanced Regression Techniques
4.3.1 Logistic Regression
Binary Outcomes: Modeling probability of binary outcomes.
Logit Function: Definition and interpretation.
Odds Ratios: Calculation and interpretation.
Model Fitting: Maximum likelihood estimation.
Goodness of Fit: Likelihood-ratio tests, Hosmer-Lemeshow test.
4.3.2 Polynomial Regression
Higher Order Terms: Including polynomial terms in regression.
Model Building: Selecting the degree of the polynomial.
Interpretation: Understanding coefficients in polynomial terms.
Overfitting: Balancing model complexity and predictive accuracy.

UNIT-III	**Machine Learning Algorithms for Predictive Modeling, Time Series Analysis**	**08 Hours**

5.1 Overview of Machine Learning
Definition and Types: Supervised, Unsupervised, Semi-supervised, and Reinforcement Learning.
Applications and Use Cases: Various domains like finance, healthcare, marketing, etc.
Key Concepts and Terminology: Features, labels, training and testing data, model training, overfitting, underfitting.
Evaluation Metrics: Accuracy, precision, recall, F1 score, ROC-AUC.
5.2 Supervised Learning Algorithms
5.2.1 Decision Trees
Structure and Components: Nodes, branches, leaves.
Splitting Criteria: Gini impurity, information gain, entropy.
Pruning: Avoiding overfitting, post-pruning and pre-pruning techniques.
Advantages and Limitations: Interpretability, complexity, overfitting issues.

5.2.2 Random Forests
Ensemble of Trees: Concept of bagging, how random forests combine multiple decision trees.
Bootstrap Aggregating: Sampling with replacement.
Feature Selection: Random selection of features for splitting.
Out-of-Bag Error: Estimating the error of the model.
Advantages and Limitations: Robustness, handling missing values, interpretability.
5.2.3 Support Vector Machines (SVM)
Hyperplane and Margin: Definition, support vectors, and maximum margin.
Kernels: Linear, polynomial, radial basis function (RBF).
Regularization Parameter (C): Controlling the trade-off between margin maximization and error minimization.
Advantages and Limitations: Effectiveness in high-dimensional spaces, computational efficiency, choice of kernel.
5.2.4 k-Nearest Neighbors (k-NN)
Algorithm Mechanics: Distance metrics (Euclidean, Manhattan).
Choosing k: Impact on model performance.
Advantages and Limitations: Simplicity, non-parametric nature, computational cost, curse of dimensionality.
5.3 Unsupervised Learning Algorithms
5.3.1 Clustering
K-means Clustering
Algorithm Steps: Initialization, assignment, update steps.
Choosing k: Elbow method, silhouette analysis.
Advantages and Limitations: Simplicity, scalability, sensitivity to initialization.
Hierarchical Clustering
Agglomerative vs Divisive: Bottom-up and top-down approaches.
Linkage Criteria: Single, complete, average.
Dendrogram: Interpretation and cutting.
Advantages and Limitations: Dendrogram interpretability, computational cost.
5.3.2 Association Rule Mining
Apriori Algorithm: Frequent itemsets, support, confidence, lift.
Applications: Market basket analysis, cross-selling strategies.
Advantages and Limitations: Rule discovery, computational complexity.
5.4 Ensemble Methods
5.4.1 Bagging and Boosting
Bagging (Bootstrap Aggregating)
Concept and Benefits: Reducing variance, improving stability.
Implementation: Bagged decision trees.
Boosting
Concept and Benefits: Reducing bias, combining weak learners.
Algorithms: AdaBoost, basic principles.
Advantages and Limitations: Improved accuracy, risk of overfitting.
5.4.2 Gradient Boosting Machines (GBM)
Concept and Mechanics: Building models sequentially, residual errors.
Algorithm Steps: Initialization, learning rate, number of trees.
Regularization Techniques: Shrinkage, subsampling.
Advantages and Limitations: High accuracy, interpretability, computational cost.

6.1 Introduction to Time Series Data
Definition and Importance: Understanding time series data and its significance in predictive analytics.
Components of Time Series: Trend, seasonality, cyclic patterns, and irregular components.
Time Series Plots: Visualization techniques for time series data.
Applications: Forecasting in finance, economics, supply chain management, meteorology, and more.
6.2 Time Series Decomposition
Additive and Multiplicative Models: Understanding the difference between additive and multiplicative decomposition.
Trend Extraction: Identifying and extracting the underlying trend component.
Seasonal Adjustment: Techniques for identifying and adjusting seasonal effects.
Residual Analysis: Examining the irregular component after trend and seasonality removal.
Practical Applications: Using decomposition to better understand and model time series data.
6.3 Smoothing Techniques
Purpose of Smoothing: Reducing noise to reveal important patterns.
6.3.1 Moving Average
Simple Moving Average (SMA): Calculation and interpretation.
Weighted Moving Average (WMA): Giving different weights to data points.
Applications: Smoothing time series data to identify trends.
6.3.2 Exponential Smoothing
Simple Exponential Smoothing: Applying smoothing parameters and exponential weights.
Holt's Linear Trend Model: Handling linear trends in time series data.
Holt-Winters Seasonal Model: Extending Holt's model to account for seasonality.
Comparison and Applications: Choosing appropriate smoothing techniques for different types of time series data.
6.4 Time Series Forecasting Models
Overview of Forecasting: Importance of accurate forecasting in various domains.
6.4.1 ARIMA (AutoRegressive Integrated Moving Average)
ARIMA Model Structure: Understanding autoregressive (AR), differencing (I), and moving average (MA) components.
Model Identification: Using ACF (autocorrelation function) and PACF (partial autocorrelation function) plots to identify appropriate parameters.
Parameter Estimation: Estimating ARIMA model parameters using statistical software.
Model Diagnostics: Checking residuals to ensure a good fit.
Applications and Examples: Real-world scenarios where ARIMA is applied.
6.4.2 Seasonal Decomposition of Time Series (STL)
STL Decomposition: Breaking down time series data into seasonal, trend, and residual components.
Seasonal Adjustment Techniques: Handling complex seasonal patterns with STL.
Model Implementation: Using STL for robust forecasting in seasonal data.
Case Studies and Examples: Practical applications of STL in various industries.

UNIT-IV	**Model Evaluation and Validation, Model Deployment**	**07 Hours**

7.1 Model Evaluation Metrics
Importance of Evaluation: Understanding why model evaluation is crucial in predictive analytics.
7.1.1 Accuracy, Precision, Recall, F1-Score
Accuracy: Definition, calculation, and limitations.
Precision: Definition, calculation, and interpretation.

Recall: Definition, calculation, and significance.
F1-Score: Combining precision and recall for a balanced evaluation metric.
Application and Examples: Practical scenarios for each metric, particularly in classification problems.
7.1.2 ROC-AUC
ROC Curve (Receiver Operating Characteristic Curve): Plotting true positive rate against false positive rate.
AUC (Area Under the Curve): Interpreting the AUC value for model performance.
Use Cases: Situations where ROC-AUC is particularly useful.
7.2 Cross-Validation Techniques
Purpose of Cross-Validation: Ensuring model generalizability and robustness.
7.2.1 k-Fold Cross-Validation
Methodology: Splitting data into k subsets, training on k-1 and validating on the remaining one.
Choosing k: Guidelines for selecting the appropriate number of folds.
Advantages and Disadvantages: Balancing bias-variance trade-off and computational efficiency.
Practical Implementation: Step-by-step guide on applying k-fold cross-validation.
7.2.2 Leave-One-Out Cross-Validation (LOOCV)
Methodology: Using each data point as a single validation case.
Comparison with k-Fold: Pros and cons of LOOCV vs. k-Fold.
When to Use LOOCV: Specific scenarios where LOOCV is beneficial.
7.3 Overfitting and Underfitting
Understanding Overfitting: Recognizing when a model is too complex and tailored to training data.
Symptoms and Consequences: How overfitting affects model performance on unseen data.
Prevention Techniques: Regularization, pruning, and other methods to mitigate overfitting.
Understanding Underfitting: Recognizing when a model is too simple to capture underlying patterns.
Symptoms and Consequences: How underfitting affects model accuracy and generalizability.
Improvement Techniques: Increasing model complexity, feature engineering, and other strategies to reduce underfitting.
7.4 Model Selection Criteria
Importance of Model Selection: Choosing the right model for the best predictive performance.
7.4.1 AIC (Akaike Information Criterion)
Definition and Calculation: Understanding AIC and its role in model selection.
Interpreting AIC Values: How to use AIC for comparing models.
Applications: Situations where AIC is most effective.
7.4.2 BIC (Bayesian Information Criterion)
Definition and Calculation: Understanding BIC and its role in model selection.
Interpreting BIC Values: Differences between AIC and BIC, and when to use BIC.
Applications: Scenarios where BIC provides more reliable model selection.

8.1 Preparing Models for Deployment
Model Export and Serialization
Formats: Overview of common formats (e.g., PMML, ONNX, pickle).
Serialization Techniques: How to serialize models for deployment.
Best Practices: Ensuring reproducibility and integrity of models during export.
Environment Setup
Infrastructure Requirements: Understanding hardware and software needs.
Deployment Platforms: Overview of cloud services (e.g., AWS SageMaker, Google AI Platform) and

on-premise options.
Dependencies Management: Handling libraries, packages, and version control.
8.2 Model Implementation
Integrating Models with Applications
APIs and Microservices: Creating and deploying REST APIs for model inference.
Frameworks: Introduction to frameworks like Flask, FastAPI, and Django for serving models.
Use Cases and Examples: Practical implementation scenarios in different industries.
Batch vs. Real-Time Processing
Batch Processing: Techniques for handling large-scale batch predictions.
Real-Time Inference: Requirements and strategies for real-time prediction serving.
Case Studies: Examples of batch and real-time deployments.
8.3 Monitoring Model Performance
Performance Metrics
Operational Metrics: Latency, throughput, and resource utilization.
Prediction Quality: Ongoing assessment of accuracy, precision, recall, and other relevant metrics.
Monitoring Tools
Logging and Alerts: Implementing logging systems and alert mechanisms for model monitoring.
Visualization Dashboards: Using tools like Grafana, Kibana, or custom dashboards to visualize performance metrics.
Drift Detection
Concept Drift: Understanding and detecting shifts in data distributions over time.
Methods for Drift Detection: Techniques like statistical tests, monitoring metrics, and control charts.
Handling Drift: Strategies to address drift in model performance.
8.4 Updating and Retraining Models
Retraining Strategies
Scheduled Retraining: Regular intervals for model retraining.
Triggered Retraining: Based on performance degradation or drift detection.
Automated Retraining Pipelines
Continuous Integration/Continuous Deployment (CI/CD): Setting up CI/CD pipelines for automated retraining and deployment.
Tools and Frameworks: Utilizing tools like Jenkins, GitLab CI, and MLflow for automation.
Version Control
Model Versioning: Managing and tracking different versions of models.
Rollback Mechanisms: Ensuring quick recovery to previous versions if new models underperform.
Post-Deployment Evaluation
A/B Testing: Implementing A/B tests to compare new models with existing ones.
Canary Deployment: Gradual rollout of new models to minimize risk.
Feedback Loops: Incorporating user feedback and new data into model improvement processes.

UNIT-V	**Applications of Predictive Analytics**	**08 Hours**

9.1 Business Analytics
Sales Forecasting
Techniques and Models: Time series analysis, regression models.
Case Studies: Real-world examples of sales forecasting.
Supply Chain Optimization
Predictive Models: Demand forecasting, inventory optimization.

Applications: Case studies from manufacturing and retail industries.
Risk Management
Fraud Detection: Techniques for identifying and predicting fraudulent activities.
Credit Scoring: Predictive models for assessing credit risk.
9.2 Financial Forecasting
Stock Market Prediction
Models: Time series analysis, machine learning techniques.
Data Sources: Financial statements, market data, news sentiment.
Portfolio Management
Risk Assessment: Predictive models for portfolio risk and return analysis.
Optimization Techniques: Techniques for optimizing investment portfolios.
Economic Forecasting
Macroeconomic Indicators: Using predictive analytics to forecast economic trends.
Case Studies: Examples of economic forecasting in government and finance sectors.
9.3 Marketing Campaigns and Customer Segmentation
Customer Segmentation
Clustering Techniques: K-means, hierarchical clustering, DBSCAN.
Applications: Segmenting customers based on purchasing behavior and demographics.
Marketing Campaign Optimization
Response Modeling: Predictive models to optimize marketing efforts and maximize ROI.
Case Studies: Examples of successful marketing campaign optimizations.
Customer Lifetime Value (CLV)
Predictive Models: Techniques for calculating and predicting CLV.
Applications: Using CLV to inform business strategies and marketing efforts.
9.4 Healthcare Analytics
Predictive Models in Healthcare
Disease Prediction: Using machine learning to predict disease onset and progression.
Patient Readmission Rates: Models to predict hospital readmissions and improve patient outcomes.
Personalized Medicine
Genomics and Predictive Analytics: Predicting patient responses to treatments based on genetic data.
Case Studies: Examples from personalized medicine and treatment plans.
Operational Efficiency
Resource Allocation: Predictive models for optimizing staffing and resource allocation in hospitals.
Case Studies: Real-world applications in healthcare management.
9.5 Web Usage Mining and E-commerce Data Analysis
Web Usage Mining
Techniques: Analyzing web logs, user behavior, and session patterns.
Applications: Improving website usability and user experience.
Recommendation Systems
Collaborative Filtering: Techniques and algorithms for recommending products.
Content-Based Filtering: Personalizing recommendations based on user preferences and item characteristics.
Customer Behavior Analysis
E-commerce Data Analysis: Analyzing purchase patterns, cart abandonment, and conversion rates.
Case Studies: Examples from leading e-commerce platforms and online retailers.
A/B Testing and Multivariate Testing
Experiment Design: Setting up and running experiments to test changes on websites.

Analysis: Interpreting results to make data-driven decisions.

TEXT BOOK:

1. Applied Predictive Analytics, Principles and Techniques for the Professional Data Analyst by Dean Abbott, 2014.

Reference Books:

1. "Data Mining: Concepts and Techniques" by Jiawei Han, Micheline Kamber, and Jian Pei
   A comprehensive guide to data mining concepts, techniques, and applications.
2. "Applied Predictive Modeling" by Max Kuhn and Kjell Johnson
   An in-depth exploration of predictive modeling techniques with practical examples.
3. "Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy
   A detailed book on machine learning algorithms and their probabilistic foundations.
4. "Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani
   A beginner-friendly introduction to statistical learning and predictive modeling.
5. "Time Series Analysis and Its Applications: With R Examples" by Robert H. Shumway and David S. Stoffer
   A practical guide to time series analysis using R.
6. "Pattern Recognition and Machine Learning" by Christopher M. Bishop
   A comprehensive textbook on pattern recognition and machine learning techniques.
7. "Practical Data Science with R" by Nina Zumel and John Mount
   A hands-on guide to data science and predictive modeling using R.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO’s-PO’s & PSO’s MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 604: PROFESSIONAL PRACTICE, LAW & ETHICS
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Understand the types of roles they are expected to play in the society as practitioners of the engineering profession.
- To develop some ideas of the legal and practical aspects of their profession.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Practice ethics and rule of the land in their profession
- Follow the principles and elements of legal contracts
- Able to resolve disputes pertaining to arbitration, reconciliation
- Aware of intellectual property loss

Course Contents

UNIT-I	Professional Practice and Ethics	07 Hours

Definition of Ethics, Professional Ethics - Engineering Ethics, Personal Ethics; Code of Ethics - Profession, Professionalism, Professional Responsibility, Conflict of Interest, Gift Vs Bribery, Environmental breaches, Negligence, Deficiencies in state-of-the-art; Vigil Mechanism, Whistle blowing, protected disclosures. Introduction to GST- Various Roles of Various Stake holders

UNIT-II	Law of Contract	08 Hours

Nature of Contract and Essential elements of valid contract, Offer and Acceptance, Consideration, Capacity to contract and Free Consent, Legality of Object. Unlawful and illegal agreements, Contingent Contracts, Performance and discharge of Contracts, Remedies for breach of contract. Contracts-II: Indemnity and guarantee, Contract of Agency, Sale of goods Act -1930: General Principles, Conditions & Warranties, Performance of Contract of Sale.

UNIT-III	Arbitration, Conciliation and ADR (Alternative Dispute Resolution) system	08 Hours

Arbitration – meaning, scope and types – distinction between laws of 1940 and 1996; UNCITRAL model law – Arbitration and expert determination; Extent of judicial intervention; International commercial arbitration.

UNIT-IV	Arbitration agreements	07

		Hours
essential and kinds, validity, reference and interim measures by court; Arbitration tribunal — appointment, challenge, jurisdiction of arbitral tribunal, powers, grounds of challenge, procedure and court assistance; Distinction between conciliation, negotiation, mediation and arbitration, confidentiality, resort to judicial proceedings, costs; Dispute Resolution Boards; Lok Adalats.		
UNIT-V	**Law relating to Intellectual property**	**08 Hours**
Introduction – meaning of intellectual property, main forms of IP, Copyright, Trademarks, Patents and Designs, Secrets; Law relating to Copyright in India including Historical evolution of Copy Rights Act, 1957, Meaning of copyright – computer programs, Ownership of copyrights and assignment, Criteria of infringement, Piracy in Internet – Remedies and procedures in India; Law relating to Patents under Patents Act, 1970		

TEXT BOOKS:

1. Professional Ethics: R. Subramanian, Oxford University Press, 2015.
2. Ravinder Kaur, Legal Aspects of Business, 4e, Cengage Learning, 2016.

REFERENCE BOOKS:

1. Wadhera (2004), Intellectual Property Rights, Universal Law Publishing Co.
2. T. Ramappa (2010), Intellectual Property Rights Law in India, Asia Law House.
3. O.P. Malhotra, Law of Industrial Disputes, N.M. Tripathi Publishers.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 621PE: SOFTWARE TESTING METHODOLOGIES
(Professional Elective – II)
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Software Engineering

Course Objective:

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using the latest tools.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand purpose of testing and path testing
- Understand strategies in data flow testing and domain testing
- Develop logic-based test strategies
- Understand graph matrices and its applications
- Implement test cases using any testing automation tool

Course Contents

UNIT-I	testing	07 Hours

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT-II	Transaction Flow Testing	08 Hours

transaction flows, transaction flow testing techniques.
Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.
Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT-III	Paths, Path products and Regular expressions	08 Hours

path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

UNIT-IV	State, State Graphs and Transition testing	07

		Hours
state graphs, good & bad state graphs, state testing, Testability tips.		
UNIT-V	Graph Matrices and Application	08 Hours
Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).		

TEXT BOOKS:

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

REFERENCE BOOKS:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 622PE: INFORMATION RETRIEVAL SYSTEMS
(Professional Elective – II)
B.Tech. III Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

- **Course Prerequisites: Data Structures**

Course Objective:

- To learn the concepts and algorithms in Information Retrieval Systems
- To understand the data/file structures that are necessary to design, and implement information retrieval (IR) systems.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Ability to apply IR principles to locate relevant information large collections of data
- Ability to design different document clustering algorithms
- Implement retrieval systems for web search tasks.
- Design an Information Retrieval System for web search tasks.

Course Contents

UNIT-I	Introduction to Information Retrieval Systems	07 Hours

Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses Information Retrieval System Capabilities: Search Capabilities, Browse Capabilities, Miscellaneous Capabilities

UNIT-II	Cataloging and Indexing	08 Hours

History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction Data Structure: Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models.

UNIT-III	Automatic Indexing	08 Hours

Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages
Document and Term Clustering: Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters

UNIT-IV	User Search Techniques	07 Hours

Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext Information Visualization: Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies		
UNIT-V	**Text Search Algorithms**	**08 Hours**
Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems Multimedia Information Retrieval: Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Imagery Retrieval, Video Retrieval		

TEXT BOOK:

1. Information Storage and Retrieval Systems – Theory and Implementation, Second Edition, Gerald J. Kowalski, Mark T. Maybury, Springer

REFERENCE BOOKS:

1. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms,
2. Prentice Hall, 1992.
3. Information Storage & Retrieval By Robert Korfhage – John Wiley & Sons.
4. Modern Information Retrieval By Yates and Neto Pearson Education.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 623PE: PATTERN RECOGNITION (Professional Elective – II)
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

- **Course Prerequisites:** Programming for problem solving
- Computer Oriented Statistical Methods

Course Objective:

- Introducing fundamental concepts, theories, and algorithms for pattern recognition and machine learning.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand the importance of pattern recognition and its representation
- Analyza the variants of NN algorithm
- Understand the necessity of Hidden markov models, decision tree and SVM for classification
- Understand different types of clustering algorithms

Course Contents

UNIT-I	Introduction: Pattern Recognition	07 Hours

Data Sets for Pattern Recognition, Different Paradigms for Pattern Recognition. Representation: Data Structures for Pattern Representation, Representation of Clusters, Proximity Measures, Size of Patterns, Abstractions of the Data Set, Feature Extraction, Feature Selection, Evaluation of Classifier, Evaluation of Clustering.

UNIT-II	Nearest Neighbor Based Classifier	08 Hours

Nearest Neighbor Algorithm, Variants of the NN Algorithm, use of the Nearest Neighbor Algorithm for Transaction Databases, Efficient Algorithms, Data Reduction, Prototype Selection. Bayes Classifier: Bayes Theorem, Minimum Error Rate Classifier, Estimation of Probabilities, Comparison with the NNC, Naïve Bayes Classifier, Bayesian Belief Network.

UNIT-III	Hidden Markov Models	08 Hours

Markov Models for Classification, Hidden Morkov Models, Classification using HMMs. Decision Trees: Introduction, Decision Tree for Pattern Classification, Construction of Decision Trees, Splitting at the Nodes, Overfitting and Pruning, Examples of Decision Tree Induction.

UNIT-IV	Support Vector Machines	07

		Hours
Introduction, Learning the Linear Discriminant Functions, Neural Networks, SVM for Classification. Combination of Classifiers: Introduction, Methods for Constructing Ensembles of Classifiers, Methods for Combining Classifiers.		
UNIT-V	**Clustering**	**08 Hours**
Importance of clustering, Hierarchical Algorithms, Partitional Clustering, Clustering Large Data Sets. An Application-Hand Written Digit Recognition: Description of the Digit Data, Preprocessing of Data, Classification Algorithms, Selection of Representative Patterns, Results.		

TEXT BOOK:

1. Pattern Recognition: An Algorithmic Approach: Murty, M. Narasimha, Devi, V. Susheela, Spinger Pub, 1st Ed.

REFERENCE BOOKS:

1. Machine Learning - Mc Graw Hill, Tom M. Mitchell.
2. Fundamentals Of Speech Recognition: Lawrence Rabiner and Biing- Hwang Juang. PrenticeHall Pub.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 624PE: COMPUTER VISION AND ROBOTICS
(Professional Elective – II)
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Linear Algebra and Probability.

Course Objective:

- To understand the Fundamental Concepts Related To sources, shadows and shading
- To understand the The Geometry of Multiple Views

Learning Course Outcome:
After successful completion of the course, students will able to:

- Implement fundamental image processing techniques required for computer vision
- Implement boundary tracking techniques
- Apply chain codes and other region descriptors, Hough Transform for line, circle, and ellipse detections.
- Apply 3D vision techniques and Implement motion related techniques.
- Develop applications using computer vision techniques.

Course Contents

UNIT-I	CAMERAS: Pinhole Cameras	07 Hours

Radiometry – Measuring Light: Light in Space, Light Surfaces, Important Special Cases
Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Interreflections: Global Shading Models
Color: The Physics of Color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.

UNIT-II	Linear Filters	08 Hours

Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates
Edge Detection: Noise, Estimating Derivatives, Detecting Edges
Texture: Representing Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.

UNIT-III	The Geometry of Multiple Views: Two Views	08 Hours

Stereopsis: Reconstruction, Human Stereposis, Binocular Fusion, Using More Cameras
Segmentation by Clustering: Segmentation, Human Vision: Grouping and Getstalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT-IV	Model and Camera Calibration	07 Hours

Segmentation by Fitting a Model: The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem, Robustness
Geometric Camera Models: Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations
Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into Account, Analytical Photogrammetry, An Application: Mobile Robot Localization

UNIT-V	Introduction to Robotics	08 Hours

Social Implications of Robotics, Brief history of Robotics, Attributes of hierarchical paradigm, Closed world assumption and frame problem, Representative Architectures, Attributes of Reactive Paradigm, Subsumption Architecture, Potential fields and Perception
Common sensing techniques for Reactive Robots: Logical sensors, Behavioural Sensor Fusion, Pro- prioceptive sensors, Proximity Sensors, Topological Planning and Metric Path Planning

TEXT BOOKS:

1. David A. Forsyth and Jean Ponce: Computer Vision — A Modern Approach, PHI Learning (Indian Edition), 2009.
2. Robin Murphy, Introduction to AI Robotics, MIT Press

REFERENCE BOOKS:

1. E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013.
2. The Robotics premier, Maja J Matari, MIT Press
3. Richard Szeliski "Computer Vision: Algorithms and Applications" Springer-Verlag London Limited 2011.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 625PE: DATA WAREHOUSING AND BUSINESS INTELLIGENCE
(Professional Elective – II)
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- This course is concerned with extracting data from the information systems that deal with the day-to-day operations and transforming it into data that can be used by businesses to drive high-level decision making
- Students will learn how to design and create a data warehouse, and how to utilize the process of extracting, transforming, and loading (ETL) data into data warehouses.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand architecture of data warehouse and OLAP operations.
- Understand Fundamental concepts of BI
- Application of BI Key Performance indicators
- Understand Utilization of Advanced BI Tools and their Implementation.
- Implementation of BI Techniques and BI Ethics.

Course Contents

UNIT-I	Data Warehouse	07 Hours

Data Warehouse Modelling, OLAP operations, Data Qube Computation methods

UNIT-II	Business Intelligence Introduction	08 Hours

Definition, Leveraging Data and Knowledge for BI, BI Components, BI Dimensions, Information Hierarchy, Business Intelligence and Business Analytics. BI Life Cycle. Data for BI - Data Issues and Data Quality for BI.

UNIT-III	BI Implementation	08 Hours

Key Drivers, Key Performance Indicators and Performance Metrics, BI Architecture/Framework, Best Practices, Business Decision Making, Styles of BI-vent-Driven alerts-A cyclic process of Intelligence Creation. The value of Business Intelligence-Value driven and Information use.

UNIT-IV	Advanced BI	07 Hours
Big Data and BI, Social Networks, Mobile BI, emerging trends, Description of different BI-Tools (Pentaho, KNIME)		
UNIT-V	**Business Intelligence and integration implementation**	**08 Hours**
connecting in BI systems- Issues of legality- Privacy and ethics- Social networking and BI.		

TEXT BOOKS:

1. Data Mining — Concepts and Techniques - JIAWEI HAN & MICHELINE KAMBER, Elsevier, 4th Edition.
2. Rajiv Sabherwal "Business Intelligence" Wiley Publications, 2012.

REFERENCE BOOKS:

1. Efraim Turban, Ramesh Sharda, Jay Aronson, David King, Decision Support and Business Intelligence Systems, 9th Edition, Pearson Education, 2009.
2. David Loshin, Business Intelligence - The Savy Manager's Guide Getting Onboard with Emerging IT, Morgan Kaufmann Publishers, 2009.
3. Philo Janus, Stacia Misner, Building Integrated Business Intelligence. Solutions with SQL Server, 2008 R2 & Office 2010, TMH, 2011.
4. Business Intelligence Data Mining and Optimization for decision making [Author: Carlo-Verellis] [Publication: (Wiley)]
5. Data Warehousing, Data Mining & OLAP- Alex Berson and Stephen J. Smith- Tata McGraw- Hill Edition, Tenth reprint 2007
6. Building the Data Warehouse- W. H. Inmon, Wiley Dreamtech India Pvt. Ltd.
7. Data Mining Introductory and Advanced topics – Margaret H Dunham, PEA.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 611OE: FUNDAMENTALS of AI (Open Elective – I)
B.Tech. III Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- To learn the difference between optimal reasoning Vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI namely, Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

Learning Course Outcome:
After successful completion of the course, students will able to:

- Gain the knowledge of what is AI, risks and benefits of AI, limits of AI and the ethics involved in building an AI application.
- Understand the nature of environments and the structure of agents.
- Possess the ability to select a search algorithm for a problem and characterize its time and space complexities.
- Possess the skill for representing knowledge using the appropriate technique
- Gain an understanding of the applications of AI

Course Contents

UNIT-I	Foundations of AI	07 Hours

Introduction to AI, History of AI, Strong and Weak AI, The State of the Art, Risks and Benefits of AI

Philosophy, Ethics and Safety of AI: The Limits of AI, Machine thinking capability, The Ethics of AI **Intelligent Agents:** Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

UNIT-II	Solving Problems by Searching	08 Hours

Problem – Solving Agents
Uninformed Search Strategies: Best-First Search, Breadth-First Search, Uniform-Cost Search, Depth-First Search, Iterative Deepening Search and Bidirectional Search
Informed Search Strategies: Greedy Best-First Search, A* Search.

UNIT-III	Order Logic	08 Hours

Logical Agents: Knowledge-based agents, Propositional Logic, Propositional Theorem Proving
First-Order Logic: Syntax and Semantics of First-Order Logic
Inference in First-Order Logic: Propositional Vs. First-Order Inference, Unification and First-Order Inference, Forward Chaining, Backward Chaining
Knowledge Representation: Ontological Engineering, Categories and Objects, Events

UNIT-IV	Quantifying Uncertainty	07 Hours

Basic Probability Notation, Inference Using Full-Joint Distributions, Independence, Bayes' Rule and its Use, Naive Bayes Models
Probabilistic Reasoning: Representing Knowledge in an Uncertain Domain, The semantics of Bayesian Networks, Exact Inference in Bayesian Networks

UNIT-V	Learning from Examples	08 Hours

Forms of Learning, Supervised Learning, Learning Decision Trees, Model Selection, Linear Regression and Classification, Ensemble Learning
Natural Language Processing: Language Models, Grammar, Parsing, Complications of RealNatural Language, Natural Language Tasks
Robotics: Robots, Robot Hardware, Kind of Problems solved, Application Domains
Computer Vision: Simple Image Features, Using Computer Vision

TEXT BOOKS:

1. "Artificial Intelligence a Modern Approach", Fourth Edition, Stuart J. Russell & Peter Norvig – Pearson.

REFERENCE BOOKS:

1. "Artificial Intelligence", Elaine Rich, Kevin Knight & Shivashankar B Nair — McGraw Hill Education.
2. Artificial Intelligence, 3rd Edn, E. Rich and K. Knight (TMH)
3. Artificial Intelligence, 3rd Edn., Patrick Henny Winston, Pearson Education.
4. Artificial Intelligence, Shivani Goel, Pearson Education.
5. Artificial Intelligence and Expert systems – Patterson, Pearson Education

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 612OE: MACHINE LEARNING BASICS (Open Elective – I)
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- To introduce students to the basic concepts and techniques of Machine Learning.
- To have a thorough understanding of the Supervised and Unsupervised learning techniques
- To study the various probability based learning techniques

Learning Course Outcome:

After successful completion of the course, students will able to:

- Distinguish between, supervised, unsupervised and semi-supervised learning
- Understand algorithms for building classifiers applied on datasets of non-linearly separable classes
- Understand the principles of evolutionary computing algorithms
- Design an ensembler to increase the classification accuracy

Course Contents

UNIT-I	Types of Machine Learning	07 Hours

Learning – Types of Machine Learning – Supervised Learning – The Brain and the Neuron – Design a Learning System – Perspectives and Issues in Machine Learning – Concept Learning Task – Concept Learning as Search – Finding a Maximally Specific Hypothesis – Version Spaces and the Candidate Elimination Algorithm – Linear Discriminants: – Perceptron – Linear Separability – Linear Regression.

UNIT-II	Multi-layer Perceptron in Practice	08 Hours

Multi-layer Perceptron– Going Forwards – Going Backwards: Back Propagation Error – Multi-layer Perceptron in Practice – Examples of using the MLP – Overview – Deriving Back-Propagation

UNIT-III	Learning with Trees	08 Hours

Decision Trees – Constructing Decision Trees – Classification and Regression Trees – Ensemble Learning – Boosting – Bagging – Different ways to Combine Classifiers – Nearest Neighbor Methods – Unsupervised Learning – K means Algorithms

UNIT-IV	Support Vector Machines	07

		Hours
Evolutionary Learning – Genetic algorithms – Genetic Offspring: - Genetic Operators – Using Genetic Algorithms		
UNIT-V	**Reinforcement Learning**	**08 Hours**
Overview – Getting Lost Example Markov Chain Monte Carlo Methods – Sampling – Proposal Distribution – Markov Chain Monte Carlo — Hidden Markov Models		

TEXT BOOKS:

1. Stephen Marsland, —Machine Learning — An Algorithmic Perspective, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.

REFERENCE BOOKS:

1. Tom M Mitchell, —Machine Learning, First Edition, McGraw Hill Education, 2013.
2. Peter Flach, —Machine Learning: The Art and Science of Algorithms that Make Sense of Data‖, First Edition, Cambridge University Press, 2012.
3. Jason Bell, —Machine learning – Hands on for Developers and Technical Professionals‖, First Edition, Wiley, 2014
4. Ethem Alpaydin, —Introduction to Machine Learning 3e (Adaptive Computation and Machine Learning Series), Third Edition, MIT Press, 2014

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 602L: MACHINE LEARNING LAB
B.Tech. III Year II Sem

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: ML Practice

Course Objective:

- The objective of this lab is to get an overview of the various machine learning techniques and can demonstrate them using python.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand modern notions in predictive data analysis
- Select data, model selection, model complexity and identify the trends
- Understand a range of machine learning algorithms along with their strengths and weaknesses
- Build predictive models from data and analyze their performance

Course Contents

List of Experiments	07 Hours

Exercise 1: Introduction to Scikit-learn for Supervised Learning
Objective: To gain hands-on experience with implementing supervised learning algorithms using the Scikit-learn library in Python.

Exercise 2: Exploring Unsupervised Learning with K-Means Clustering
Objective: To explore the concepts of unsupervised learning and clustering using the K-Means algorithm.

Exercise 3: Implementing Linear Regression from Scratch
Objective: To gain a deeper understanding of linear regression by implementing it from scratch using Python.

Exercise 4: Binary Classification with Logistic Regression
Objective: To implement logistic regression for binary classification tasks and understand its application in real-world scenarios.

Exercise 5: Decision Tree Classifier for Multiclass Classification
Objective: To understand the working of decision tree classifiers and their application in multiclass classification problems.

Exercise 6: Support Vector Machine (SVM) for Classification
Objective: To implement support vector machine (SVM) classifiers for binary and multiclass classification tasks and understand their kernel trick.

Exercise 7: Ensemble Learning with Random Forests
Objective: To implement random forests, an ensemble learning method, and understand their advantages in improving model performance and generalization.

Exercise 8: Hierarchical Clustering
Objective: To understand hierarchical clustering algorithms and their application in grouping data points

based on their similarity.

Exercise 9: Density-Based Clustering (DBSCAN)

Objective: To implement the DBSCAN algorithm and understand its effectiveness in identifying clusters of arbitrary shapes in noisy datasets.

Exercise 10: Principal Component Analysis (PCA)

Objective: To understand the concept of dimensionality reduction using PCA and apply it to high-dimensional datasets for visualization and feature extraction.

Exercise 11: Convolutional Neural Network (CNN) for Image Classification

Objective: To implement a Convolutional Neural Network (CNN) using Python and TensorFlow/Keras library for image classification tasks.

Exercise 12: Recurrent Neural Network (RNN) for Text Generation

Objective: To implement a Recurrent Neural Network (RNN) using Python and TensorFlow/Keras library for text generation tasks.

Exercise 13: Reinforcement Learning with OpenAI Gym

Objective: To implement reinforcement learning algorithms using Python and OpenAI Gym library for training agents in various environments.

Exercise 14: Evaluation Metrics for Classification Models

Objective: To understand and implement various evaluation metrics for classification models using Python and Scikit-learn library.

Exercise 15: Regression Model Evaluation Techniques

Objective: To explore and apply various evaluation techniques for regression models using Python and Scikit-learn library.

Exercise 16: Cluster Validation Measures

Objective: To understand and apply cluster validation measures to assess the quality of clustering solutions using Python and Scikit-learn library.

Exercise 17: Model Selection Techniques

Objective: To explore and apply model selection techniques for choosing the best machine learning models using Python and Scikit-learn library.

Exercise 18: Hyperparameter Tuning and Optimization

Objective: To understand and apply hyperparameter tuning and optimization techniques for improving machine learning model performance using Python and Scikit-learn library.

Exercise 19: Transfer Learning in Image Classification

Objective: To apply transfer learning techniques using pre-trained convolutional neural network (CNN) models for image classification tasks using TensorFlow/Keras.

Exercise 20: Meta-Learning for Few-Shot Learning

Objective: To implement meta-learning algorithms for few-shot learning tasks using Python and TensorFlow/Keras.

Exercise 21: Adversarial Attacks and Defenses

Objective: To understand adversarial attacks and defenses in deep learning models and evaluate their robustness using Python and TensorFlow.

TEXT BOOK:

1. Machine Learning – Tom M. Mitchell, - MGH.

REFERENCE BOOK:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 603L: PREDICTIVE ANALYTICS LAB
B.Tech. III Year II Sem

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: ML Practice

Course Objective:

- The objective of this lab is to get an overview of the various machine learning techniques and can demonstrate them using python.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand modern notions in predictive data analysis
- Select data, model selection, model complexity and identify the trends
- Understand a range of machine learning algorithms along with their strengths and weaknesses
- Build predictive models from data and analyze their performance

Course Contents

List of Experiments	07 Hours

Unit 1: Introduction to Predictive Analytics

Lab 1: Introduction to Predictive Analytics Tools

- Objective: Familiarize students with Python, R, and Tidyverse for predictive analytics.
- Tasks:
  - Install and set up Python and R environments.
  - Introduction to Jupyter Notebooks and RStudio.
  - Basic operations and syntax in Python and R.

Unit 2: Data Understanding and Preparation

Lab 2: Data Collection and Importing Data

- Objective: Understand data collection methods and how to import data.
- Tasks:
  - Import datasets from CSV, Excel, and databases into Python (using Pandas) and R (using readr and DBI packages).

Lab 3: Data Cleaning and Preprocessing

- Objective: Learn data cleaning and preprocessing techniques.
- Tasks:
  - Handle missing values using Pandas in Python and Tidyverse in R.
  - Perform data transformations and normalization.

Lab 4: Exploratory Data Analysis (EDA)

- Objective: Perform EDA to understand data distributions and relationships.
- Tasks:

    - Generate descriptive statistics in Python (Pandas) and R (dplyr).
    - Create visualizations (histograms, box plots) using Matplotlib and Seaborn in Python, and ggplot2 in R.

Lab 5: Feature Engineering

- Objective: Practice feature selection and extraction techniques.
- Tasks:
    - Use Python's Scikit-learn and R's caret package for feature selection.
    - Perform feature extraction techniques such as PCA.

Unit 3: Probability Theory and Statistics for Predictive Modeling

Lab 6: Probability Distributions

- Objective: Understand and visualize different probability distributions.
- Tasks:
    - Generate and plot discrete and continuous distributions in Python (SciPy) and R (ggplot2).

Lab 7: Statistical Inference

- Objective: Perform point estimation, confidence intervals, and hypothesis testing.
- Tasks:
    - Conduct hypothesis tests and construct confidence intervals using Python (SciPy, Statsmodels) and R (stats package).

Unit 4: Regression Analysis

Lab 8: Simple Linear Regression

- Objective: Implement and evaluate simple linear regression models.
- Tasks:
    - Build and interpret a simple linear regression model in Python (Scikit-learn) and R (lm function).

Lab 9: Multiple Linear Regression

- Objective: Develop and analyze multiple linear regression models.
- Tasks:
    - Perform multiple linear regression and interpret coefficients in Python (Scikit-learn) and R (lm function).
    - Conduct diagnostics and residual analysis.

Lab 10: Advanced Regression Techniques

- Objective: Implement logistic and polynomial regression models.
- Tasks:
    - Build logistic regression models in Python (Scikit-learn) and R (glm function).
    - Create polynomial regression models and evaluate performance.

Unit 5: Machine Learning Algorithms for Predictive Modeling

Lab 11: Decision Trees

- Objective: Implement and evaluate decision tree models.
- Tasks:
    - Train decision tree models using Python (Scikit-learn) and R (rpart package).
    - Visualize and interpret decision trees.

Lab 12: Random Forests

- Objective: Develop and analyze random forest models.
- Tasks:
    - Build random forest models in Python (Scikit-learn) and R (randomForest package).

- Evaluate model performance using accuracy and feature importance.

Lab 13: Support Vector Machines (SVM)

- Objective: Implement SVM for classification tasks.
- Tasks:
  - Train SVM models in Python (Scikit-learn) and R (e1071 package).
  - Tune hyperparameters and evaluate model performance.

Lab 14: k-Nearest Neighbors (k-NN)

- Objective: Apply k-NN algorithm for classification and regression.
- Tasks:
  - Implement k-NN in Python (Scikit-learn) and R (class package).
  - Optimize the value of k and assess model accuracy.

Lab 15: Clustering Algorithms

- Objective: Perform clustering using k-means and hierarchical clustering.
- Tasks:
  - Apply k-means clustering in Python (Scikit-learn) and R (stats package).
  - Implement hierarchical clustering and visualize dendrograms.

Lab 16: Association Rule Mining

- Objective: Discover association rules in datasets.
- Tasks:
  - Use Python (mlxtend package) and R (arules package) to perform association rule mining.

Lab 17: Ensemble Methods

- Objective: Implement ensemble learning techniques.
- Tasks:
  - Apply bagging and boosting methods using Python (Scikit-learn) and R (caret package).
  - Build Gradient Boosting Machines (GBM) and evaluate performance.

Unit 6: Time Series Analysis

Lab 18: Time Series Decomposition

- Objective: Decompose time series data into trend, seasonality, and residuals.
- Tasks:
  - Perform time series decomposition in Python (Statsmodels) and R (stl function).

Lab 19: Smoothing Techniques

- Objective: Apply smoothing techniques to time series data.
- Tasks:
  - Implement moving average and exponential smoothing in Python (Pandas) and R (forecast package).

Lab 20: Time Series Forecasting Models

- Objective: Develop time series forecasting models.
- Tasks:
  - Build ARIMA models in Python (Statsmodels) and R (forecast package).
  - Apply Seasonal Decomposition of Time Series (STL) and forecast.

Unit 7: Model Evaluation and Validation

Lab 21: Model Evaluation Metrics

- Objective: Calculate and interpret model evaluation metrics.
- Tasks:

  - Compute accuracy, precision, recall, F1-score, and ROC-AUC in Python (Scikit-learn) and R (caret package).

Lab 22: Cross-Validation Techniques

- Objective: Implement cross-validation methods.
- Tasks:
  - Perform k-fold and leave-one-out cross-validation in Python (Scikit-learn) and R (caret package).

Lab 23: Overfitting and Underfitting

- Objective: Identify and mitigate overfitting and underfitting.
- Tasks:
  - Visualize model performance to detect overfitting and underfitting.
  - Apply regularization techniques (Lasso, Ridge) in Python (Scikit-learn) and R (glmnet package).

Lab 24: Model Selection Criteria

- Objective: Use AIC and BIC for model selection.
- Tasks:
  - Calculate AIC and BIC for model comparison in Python (Statsmodels) and R (stats package).

Unit 8: Model Deployment

Lab 25: Preparing Models for Deployment

- Objective: Prepare predictive models for deployment.
- Tasks:
  - Export trained models using Python (joblib, pickle) and R (saveRDS, PMML).

Lab 26: Model Implementation

- Objective: Implement models in a production environment.
- Tasks:
  - Deploy models as web services using Flask (Python) and plumber (R).

Lab 27: Monitoring and Updating Models

- Objective: Monitor and update deployed models.
- Tasks:
  - Set up monitoring for model performance metrics.
  - Retrain models with new data and update deployment using Python (Scikit-learn, Flask) and R (caret, plumber).

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 605L: UI DESIGN-FLUTTER
B.Tech. III Year II Sem

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Learns to Implement Flutter Widgets and Layouts
- Understands Responsive UI Design and with Navigation in Flutter
- Knowledge on Widges and customize widgets for specific UI elements, Themes
- Understand to include animation apart from fetching data

Learning Course Outcome:
After successful completion of the course, students will able to:

- Implements Flutter Widgets and Layouts
- Responsive UI Design and with Navigation in Flutter
- Create custom widgets for specific UI elements and also Apply styling using themes and custom styles.
- Design a form with various input fields, along with validation and error handling
- Fetches data and write code for unit Test for UI components and also animation

Course Contents

List of Experiments	07 Hours

Students need to implement the following experiments

1. a) Install Flutter and Dart SDK.
   b) Write a simple Dart program to understand the language basics.

2. a) Explore various Flutter widgets (Text, Image, Container, etc.).
   b) Implement different layout structures using Row, Column, and Stack widgets.

3. a) Design a responsive UI that adapts to different screen sizes.
   b) Implement media queries and breakpoints for responsiveness.

4. a) Set up navigation between different screens using Navigator.
   b) Implement navigation with named routes.

5. a) Learn about stateful and stateless widgets.
   b) Implement state management using set State and Provider.

6. a) Create custom widgets for specific UI elements.
   b) Apply styling using themes and custom styles.

7. a) Design a form with various input fields.
   b) Implement form validation and error handling.

8. a) Add animations to UI elements using Flutter's animation framework.
   b) Experiment with different types of animations (fade, slide, etc.).

9. a) Fetch data from a REST API.
   b) Display the fetched data in a meaningful way in the UI.

10. a) Write unit tests for UI components.
    b) Use Flutter's debugging tools to identify and fix issues.

TEXT BOOK:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 701: BIG DATA ANALYTICS
B.Tech. III Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Understand the fundamental concepts and characteristics of big data and its importance in modern data-driven decision-making.
- Explore the architecture and components of big data platforms like Hadoop and Spark.
- Develop skills in data collection, storage, and processing techniques specific to big data.
- Learn about various big data storage solutions, including NoSQL databases.
- Master techniques for data cleaning, transformation, and preparation for analysis.
- Apply advanced analytical methods and machine learning techniques to big data.
- Gain proficiency in using big data tools and frameworks for real-world data analysis.
- Interpret and visualize large data sets effectively to communicate insights.
- Analyze real-world case studies and understand the application of big data analytics in different industries.
- Prepare students for advanced study and careers in big data analytics and data science.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Explain the key concepts, opportunities, and challenges of big data.
- Utilize the Hadoop ecosystem and Spark framework for big data processing.
- Implement data collection, storage, and processing workflows for large datasets.
- Employ NoSQL databases for efficient data storage and retrieval.
- Perform data cleaning, transformation, and preparation techniques on big data.
- Apply machine learning algorithms and advanced analytics techniques to big data.
- Use big data tools such as Hadoop, Spark, and NoSQL databases for practical data analysis tasks.
- Create visualizations to effectively communicate insights derived from big data.
- Implement stream processing pipelines for real-time data analysis.
- Analyze case studies to understand the application of big data analytics in various industries.
- Demonstrate the ability to manage and analyze big data projects, preparing for roles in data science and big data analytics.

Course Contents

UNIT-I	Introduction to Big Data, Big Data Technologies	07 Hours

1.1. Overview of Big Data
- Definition and Concept
- Evolution of Big Data
- Big Data Ecosystem

1.2. Characteristics and Challenges of Big Data
- Volume, Velocity, Variety, Veracity, and Value
- Data Quality and Management Issues
- Scalability and Storage Solutions
- Security and Privacy Concerns

1.3. Big Data vs Traditional Data Analysis
- Differences in Data Processing
- Advantages of Big Data
- Limitations and Trade-offs
- Case Studies Comparing Both Approaches

1.4. Importance and Applications of Big Data Analytics
- Role in Decision-Making
- Industry-Specific Applications (Healthcare, Finance, Retail, etc.)
- Predictive and Prescriptive Analytics
- Success Stories and Real-World Examples

2.1. Introduction to Hadoop Ecosystem
- Overview of Hadoop Ecosystem Components
  - HDFS (Hadoop Distributed File System)
  - MapReduce
  - YARN (Yet Another Resource Negotiator)
  - Hadoop Common
- Role of Each Component in Big Data Processing
- Installation and Configuration of Hadoop

2.2. Hadoop Distributed File System (HDFS)
- Architecture and Design Principles
  - NameNode and DataNode
  - Block Storage Mechanism
- HDFS Operations
  - Read/Write Operations
  - Data Replication
  - Fault Tolerance and Reliability
- HDFS Shell and Command Line Interface
- Best Practices for HDFS Usage

2.3. MapReduce Programming Model
- Basic Concepts of MapReduce
  - Map and Reduce Functions
  - Data Flow and Processing
- Writing a MapReduce Program
  - Key-Value Pair Concept
  - Example Programs (Word Count, Sorting)

Advanced MapReduce Techniques
Combiner and Partitioner
Optimization and Performance Tuning
Real-World Use Cases and Applications
2.4. Introduction to Spark Framework
Overview of Apache Spark
Components and Architecture
Spark Core, Spark SQL, Spark Streaming, MLlib, GraphX
Spark vs. Hadoop: A Comparative Analysis
Programming with Spark
RDD (Resilient Distributed Dataset)
DataFrames and Datasets
Writing Spark Applications
Basic Operations (Transformations and Actions)
Example Programs (Word Count, DataFrame Operations)
2.5. Comparison of Hadoop and Spark
Performance Comparison
Batch Processing vs. Real-Time Processing
Speed and Efficiency
Usability and Flexibility
Ease of Use and Learning Curve
Programming Languages Supported (Java, Scala, Python, R)
Ecosystem and Integration
Compatibility with Other Big Data Tools
Scalability and Fault Tolerance
Cost and Resource Utilization
Resource Management and Optimization
Cost of Deployment and Maintenance

UNIT-II	Data Collection and Storage	08 Hours

3.1. Data Collection Techniques
Overview of Data Collection Methods
Structured Data Collection
Semi-Structured and Unstructured Data Collection
Data Sources
Logs and Event Data
Social Media and Web Data
Sensors and IoT Devices
Data Quality and Preprocessing
Data Cleaning
Handling Missing Values
Data Normalization and Transformation
3.2. Data Ingestion Tools (Apache Flume, Sqoop)
Introduction to Data Ingestion
Importance of Data Ingestion in Big Data

- Batch vs. Real-Time Data Ingestion
- Apache Flume
  - Architecture and Components
  - Sources, Channels, and Sinks
  - Configuring and Deploying Flume Agents
  - Example Use Cases
- Apache Sqoop
  - Introduction and Use Cases
  - Importing Data from Relational Databases to Hadoop
  - Exporting Data from Hadoop to Relational Databases
  - Performance Tuning and Optimization

3.3. Data Storage Solutions
- Overview of Big Data Storage Requirements
  - Scalability
  - Durability
  - Consistency
- Distributed File Systems
  - HDFS (Hadoop Distributed File System)
  - Google File System (GFS)
- Cloud-Based Storage Solutions
  - Amazon S3
  - Google Cloud Storage
  - Azure Blob Storage

3.4. Introduction to NoSQL Databases
- Characteristics of NoSQL Databases
  - Schema-less Data Models
  - Horizontal Scalability
  - High Availability and Fault Tolerance
- Types of NoSQL Databases
  - Key-Value Stores
  - Document Stores
  - Column-Family Stores
  - Graph Databases
- Comparison with Traditional RDBMS
  - Advantages and Disadvantages
  - Use Cases and Applications

3.5. Key-Value Stores, Document Stores, Column-Family Stores
- Key-Value Stores
  - Basic Concepts and Architecture
  - Examples: Redis, DynamoDB
  - Use Cases and Performance Considerations
- Document Stores
  - Document-Oriented Data Model
  - Examples: MongoDB, CouchDB
  - Querying and Indexing Documents
- Column-Family Stores

Data Model and Architecture Examples: Apache Cassandra, HBase Column Families and Wide-Column Data Storage Querying and Performance Tuning		
UNIT-III	**Data Processing and Management**	**08 Hours**
4.1. Data Cleaning and Preparation Importance of Data Cleaning Impact on Analysis Accuracy Data Quality Issues Data Cleaning Techniques Removing Duplicates Handling Missing Values Detecting and Correcting Errors Data Standardization and Normalization Data Preparation Data Integration and Fusion Feature Engineering Splitting Data into Training and Testing Sets Dealing with Imbalanced Datasets 4.2. Data Transformation Techniques Overview of Data Transformation Objectives and Benefits ETL (Extract, Transform, Load) Process Common Data Transformation Methods Aggregation and Summarization Filtering and Subsetting Data Enrichment Advanced Transformation Techniques Data Binning Encoding Categorical Variables Dimensionality Reduction (e.g., PCA) Text and Time Series Data Transformations 4.3. Workflow Orchestration Tools (Apache Oozie, Airflow) Introduction to Workflow Orchestration Importance in Big Data Processing Key Features of Orchestration Tools Apache Oozie Overview and Architecture Defining Workflows and Coordinators Job Scheduling and Dependency Management Oozie Workflow Language Apache Airflow Overview and Architecture Creating and Managing Workflows with DAGs (Directed Acyclic Graphs)		

Task Scheduling, Execution, and Monitoring
Integrating Airflow with Hadoop, Spark, and Other Big Data Tools
4.4. Batch Processing with Hadoop and Spark
Introduction to Batch Processing
Batch vs. Real-Time Processing
Use Cases for Batch Processing
Batch Processing with Hadoop
MapReduce Programming Model
Job Configuration
Map and Reduce Functions
Workflow Execution with YARN
Performance Optimization Techniques
Batch Processing with Spark
RDDs (Resilient Distributed Datasets) and DataFrames
Spark SQL and DataFrame API
Spark Core: Transformations and Actions
Optimizing Spark Jobs
Integrating Spark with Hive, HBase, and Other Data Sources
4.5. Real-Time Processing with Apache Storm and Kafka
Introduction to Real-Time Processing
Importance and Applications
Challenges in Real-Time Data Processing
Real-Time Processing with Apache Storm
Storm Architecture and Components (Spouts, Bolts, Topologies)
Building and Deploying Storm Topologies
Fault Tolerance and Reliability in Storm
Integrating Storm with Hadoop and HDFS
Real-Time Data Streaming with Apache Kafka
Kafka Architecture (Producers, Consumers, Brokers, Topics)
Kafka Streams API for Stream Processing
Building Real-Time Data Pipelines with Kafka Connect
Ensuring Data Durability and Scalability with Kafka
Integrating Kafka with Spark Streaming and Storm

UNIT-IV	**Query Languages and Tools, Advanced Analytical Methods**	**07 Hours**

5.1. Introduction to HiveQL
Overview of Hive
Apache Hive Architecture
Key Features and Use Cases
HiveQL Basics
Data Definition Language (DDL)
CREATE, ALTER, DROP Statements
Data Manipulation Language (DML)
SELECT, INSERT, UPDATE, DELETE Statements
Data Types and Storage Formats

Primitive and Complex Data Types
File Formats (Text, Parquet, ORC)
5.2. Querying Big Data with Hive
Writing Hive Queries
Basic Query Structure
Joins and Unions
Subqueries and Nested Queries
Aggregation Functions and Grouping
Hive Functions
Built-in Functions
String Functions
Date Functions
Mathematical Functions
User-Defined Functions (UDFs)
Creating and Using UDFs
Hive Query Performance Optimization
Partitioning and Bucketing
Indexing and Query Caching
Cost-Based Optimization (CBO)
Integrating Hive with Other Tools
Hive and HDFS
Hive and Spark SQL
Hive and HBase
5.3. Data Processing with Pig
Introduction to Pig
Pig Architecture and Execution Modes
Pig Latin Language Overview
Writing Pig Scripts
Loading and Storing Data
Relational Operators (FOREACH, FILTER, JOIN, GROUP, ORDER BY)
User-Defined Functions (UDFs) in Pig
Pig Data Types and Schemas
Scalar and Complex Data Types
Defining and Using Schemas
Advanced Pig Techniques
Data Transformations and Aggregations
Error Handling and Debugging Pig Scripts
Optimizing Pig Performance
Pig and Hadoop Integration
Running Pig on Hadoop
Pig and HCatalog Integration
5.4. Using JAQL for Data Analysis
Introduction to JAQL
Overview and Architecture
Key Features and Use Cases
JAQL Syntax and Data Model

Basic Syntax and Expressions
Handling JSON Data in JAQL
Writing JAQL Queries
Data Loading and Transformation
Filtering, Mapping, and Reducing Data
Aggregations and Grouping
Advanced JAQL Techniques
Joins and Nested Queries
User-Defined Functions in JAQL
Error Handling and Debugging
Integrating JAQL with Hadoop Ecosystem
Running JAQL on Hadoop
JAQL and HDFS Integration
JAQL with Hive and HBase
5.5. Advanced Query Optimization Techniques
Importance of Query Optimization
Impact on Performance and Resource Utilization
Optimization Strategies in Hive
Cost-Based Optimization (CBO)
Partition Pruning
Predicate Pushdown
Vectorized Query Execution
Optimizing Pig Scripts
Understanding Logical and Physical Plans
Combiner Optimization
Multi-query Execution
Schema and Type Optimization
General Optimization Techniques
Data Partitioning and Distribution
Indexing and Caching Strategies
Efficient Join and Aggregation Techniques
Monitoring and Tuning Query Performance
Analyzing Query Execution Plans
Resource Allocation and Management
Using Tools for Performance Monitoring (Ganglia, Nagios, Ambari)

6.1. Introduction to Machine Learning on Big Data
Overview of Machine Learning Concepts
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Machine Learning Challenges with Big Data
Scalability
Data Quality and Preprocessing
Model Training and Evaluation
Big Data Machine Learning Platforms

Apache Spark MLlib
Hadoop Mahout
TensorFlow on Big Data
6.2. Implementing Algorithms with Spark MLlib
Overview of Spark MLlib
Key Features and Architecture
MLlib vs. ML Pipeline API
Data Preparation for MLlib
Loading and Transforming Data
Feature Engineering and Selection
Supervised Learning Algorithms
Linear Regression
Logistic Regression
Decision Trees and Random Forests
Support Vector Machines
Unsupervised Learning Algorithms
K-means Clustering
Principal Component Analysis (PCA)
Singular Value Decomposition (SVD)
Model Evaluation and Tuning
Cross-Validation
Hyperparameter Tuning
Metrics for Model Performance
6.3. Scalable Analytics with R and Python
Introduction to Scalable Analytics
Benefits and Challenges
Key Concepts and Approaches
Using R for Big Data Analytics
R Packages for Big Data (e.g., dplyr, data.table)
Integrating R with Hadoop and Spark
Parallel Processing in R
Using Python for Big Data Analytics
Python Libraries for Big Data (e.g., NumPy, Pandas, Dask)
Integrating Python with Hadoop and Spark
Parallel Processing in Python with Dask and PySpark
Comparative Analysis
R vs. Python for Big Data
Selecting the Right Tool for the Task
6.4. Predictive Analytics and Modeling
Overview of Predictive Analytics
Concepts and Applications
Workflow for Predictive Modeling
Building Predictive Models
Data Collection and Preparation
Feature Engineering
Model Selection and Training

Techniques in Predictive Analytics
Regression Analysis
Classification Techniques
Decision Trees, Random Forests, Gradient Boosting
Time Series Analysis and Forecasting
Model Evaluation and Deployment
Evaluating Model Performance
Model Deployment Strategies
Monitoring and Updating Models
6.5. Case Studies on Advanced Analytics
Case Study 1: Predictive Maintenance
Problem Definition and Data Collection
Model Building and Evaluation
Implementation and Results
Case Study 2: Customer Segmentation
Data Preparation and Feature Selection
Clustering Analysis
Business Implications and Actionable Insights
Case Study 3: Fraud Detection
Identifying Fraudulent Patterns
Model Training and Validation
Real-time Fraud Detection System
Case Study 4: Recommendation Systems
Collaborative Filtering Techniques
Building a Recommendation Model
Evaluation and Tuning
Lessons Learned and Best Practices
Key Takeaways from Case Studies
Common Pitfalls and How to Avoid Them
Future Trends in Big Data Analytics

UNIT-V	**Data Visualization and Interpretation, Stream Processing and Real-Time Analytics**	**08 Hours**

7.1. Principles of Data Visualization
Basics of Data Visualization
Definition and Importance
History and Evolution of Data Visualization
Key Principles
Clarity and Simplicity
Accuracy and Integrity
Context and Relevance
Types of Data Visualizations
Charts: Bar, Line, Pie, Scatter Plots
Graphs and Networks
Maps and Geospatial Visualizations
Dashboards and Interactive Visualizations

Design Best Practices
Choosing the Right Visualization Type
Color Theory and Usage
Layout and Design Principles
Avoiding Common Pitfalls
7.2. Tools for Big Data Visualization (Tableau, D3.js)
Overview of Visualization Tools
Criteria for Selecting Visualization Tools
Tableau
Introduction to Tableau
Key Features and Capabilities
Data Connection and Preparation
Creating Basic Visualizations
Advanced Tableau Features (Calculated Fields, Parameters, Tableau Prep)
D3.js
Introduction to D3.js
Key Features and Capabilities
Basic D3.js Syntax and Structure
Creating Simple Visualizations with D3.js
Advanced D3.js Techniques (Transitions, Animations, Interactivity)
Comparison and Integration
Integrating Tableau and D3.js with Big Data Platforms
Strengths and Weaknesses of Each Tool
7.3. Creating Effective Visualizations
Visualization Process
Data Preparation and Cleaning
Identifying Key Insights and Messages
Selecting Appropriate Visualization Techniques
Design and Implementation
Designing for Your Audience
Using Visual Elements to Enhance Understanding
Interactive vs. Static Visualizations
Enhancing Visual Appeal
Using Color and Fonts Effectively
Incorporating Branding Elements
Creating Storyboards and Prototypes
Testing and Refinement
User Testing and Feedback
Iterative Design and Improvement
Ensuring Accessibility and Inclusivity
7.4. Interpreting and Communicating Insights
Interpreting Data Visualizations
Analyzing Patterns, Trends, and Outliers
Drawing Conclusions from Visual Data
Identifying Misleading Visualizations
Storytelling with Data

- Crafting a Narrative
- Structuring the Presentation of Insights
- Using Visuals to Support Your Story

Effective Communication Strategies
- Tailoring Communication to Different Audiences
- Combining Visuals with Verbal and Written Explanations
- Persuasive Presentation Techniques

Documentation and Reporting
- Creating Comprehensive Reports
- Including Visualizations in Reports and Presentations
- Ensuring Reproducibility and Transparency

7.5. Real-World Visualization Examples

Case Study 1: Business Analytics
- Visualizing Sales and Revenue Data
- Creating Dashboards for Business Intelligence
- Interactive Visualizations for Decision Support

Case Study 2: Healthcare Analytics
- Visualizing Patient Data and Health Outcomes
- Mapping Disease Spread and Epidemiological Data
- Using Visualizations for Public Health Communication

Case Study 3: Social Media Analytics
- Analyzing Social Media Trends and Sentiments
- Network Visualizations of Social Connections
- Interactive Visuals for Social Media Engagement

Case Study 4: Environmental Data
- Visualizing Climate Data and Environmental Changes
- Geospatial Visualizations of Environmental Impact
- Using Visualizations for Environmental Policy and Advocacy

8.1. Introduction to Stream Processing

Concepts and Definitions
- Definition of Stream Processing
- Differences Between Stream Processing and Batch Processing
- Key Terminologies: Streams, Events, Windows

Importance of Stream Processing
- Applications and Benefits
- Challenges in Stream Processing

Stream Processing Models
- Event-driven Architecture
- Data Flow Models
- Windowing and Aggregation Techniques

Stream Processing vs. Real-Time Analytics
- Understanding the Need for Real-Time Data
- Use Cases in Various Industries (Finance, Healthcare, IoT)

8.2. Tools for Real-Time Data Processing (Apache Flume, Kafka)

Apache Flume

- Overview and Architecture
- Key Components: Sources, Channels, Sinks
- Data Flow in Flume
- Configuring and Deploying Flume Agents
- Use Cases and Examples

Apache Kafka
- Overview and Architecture
- Key Concepts: Producers, Consumers, Topics, Partitions, Brokers
- Kafka Streams API
- Setting Up a Kafka Cluster
- Data Ingestion and Processing with Kafka
- Use Cases and Examples

Comparison of Flume and Kafka
- Strengths and Weaknesses
- Use Case Scenarios for Each Tool

8.3. Building Stream Processing Pipelines

Pipeline Architecture
- Designing a Stream Processing Pipeline
- Key Components and Their Roles
- Integrating Various Tools and Technologies

Data Ingestion
- Techniques for Capturing Real-Time Data
- Integrating Data Sources with Flume and Kafka
- Handling Data Streams at Scale

Stream Processing Frameworks
- Introduction to Apache Storm
  - Architecture and Key Concepts
  - Setting Up and Configuring Storm Clusters
  - Developing Storm Topologies
- Introduction to Apache Samza
  - Architecture and Key Concepts
  - Setting Up and Configuring Samza Jobs
  - Developing Samza Applications
- Introduction to Apache Flink
  - Architecture and Key Concepts
  - Setting Up and Configuring Flink Clusters
  - Developing Flink Applications

Data Transformation and Enrichment
- Applying Transformations on Streaming Data
- Joining and Aggregating Data Streams
- Handling Out-of-Order and Late Arriving Data

Monitoring and Managing Pipelines
- Tools for Monitoring Stream Processing
- Troubleshooting and Debugging Pipelines
- Scaling Stream Processing Applications

8.4. Real-Time Analytics Use Cases

Financial Services
- Fraud Detection and Prevention
- Real-Time Trading Analytics
- Customer Behavior Analysis

Healthcare
- Real-Time Patient Monitoring
- Predictive Analytics for Emergency Response
- Disease Outbreak Tracking

E-Commerce and Retail
- Real-Time Inventory Management
- Personalized Recommendations
- Customer Sentiment Analysis

Telecommunications
- Network Performance Monitoring
- Real-Time Customer Support
- Usage Pattern Analysis

Internet of Things (IoT)
- Smart Cities and Infrastructure
- Real-Time Environmental Monitoring
- Predictive Maintenance

Case Studies and Examples
- Detailed Analysis of Successful Real-Time Analytics Implementations
- Lessons Learned and Best Practices
- Future Trends in Real-Time Analytics

UNIT-VI	Case Studies and Applications	08 Hours

9.1. Big Data Analytics in Business

Overview of Big Data in Business
- Importance and Benefits of Big Data Analytics in Business
- Common Challenges and Solutions

Customer Insights and Personalization
- Analyzing Customer Behavior and Preferences
- Personalized Marketing and Recommendation Systems
- Case Study: Netflix's Recommendation Algorithm

Operational Efficiency and Optimization
- Supply Chain and Logistics Optimization
- Predictive Maintenance and Asset Management
- Case Study: UPS's ORION Routing System

Financial Analytics
- Risk Management and Fraud Detection
- Real-Time Financial Analytics and Trading
- Case Study: Credit Scoring Models in Banking

Human Resources and Talent Management
- People Analytics and Employee Retention Strategies
- Workforce Planning and Optimization

Case Study: Google's People Operations
9.2. Big Data in Healthcare
Overview of Big Data in Healthcare
Importance and Impact on Healthcare
Challenges and Ethical Considerations
Electronic Health Records (EHR)
Data Integration and Management
Predictive Analytics for Patient Outcomes
Case Study: Predictive Analytics in Hospital Readmissions
Personalized Medicine and Genomics
Analyzing Genomic Data for Personalized Treatments
Drug Discovery and Development
Case Study: IBM Watson in Oncology
Public Health and Epidemiology
Tracking and Predicting Disease Outbreaks
Health Surveillance and Population Health Management
Case Study: Google Flu Trends
Remote Monitoring and Telemedicine
Real-Time Patient Monitoring Systems
Big Data in Telehealth Services
Case Study: Remote Monitoring for Chronic Disease Management
9.3. Big Data for Social Media Analysis
Overview of Big Data in Social Media
Importance and Applications in Social Media
Data Privacy and Ethical Issues
Sentiment Analysis and Opinion Mining
Techniques for Analyzing Social Media Sentiments
Tools and Frameworks for Sentiment Analysis
Case Study: Sentiment Analysis for Brand Monitoring
Influencer Identification and Network Analysis
Identifying Key Influencers and Opinion Leaders
Social Network Analysis Techniques
Case Study: Influencer Marketing Strategies
Trend Analysis and Event Detection
Real-Time Trend Analysis on Social Media Platforms
Event Detection and Analysis
Case Study: Twitter Data Analysis for Event Detection
Campaign Performance and Social Media ROI
Measuring the Impact of Social Media Campaigns
Techniques for Calculating Social Media ROI
Case Study: Campaign Analysis for a Major Brand
9.4. Big Data in E-commerce
Overview of Big Data in E-commerce
Importance and Benefits for E-commerce Businesses
Key Challenges and Considerations
Customer Behavior and Personalization

- Analyzing Browsing and Purchase Behavior
- Personalized Recommendations and Marketing
- Case Study: Amazon's Recommendation Engine

Inventory Management and Demand Forecasting
- Optimizing Inventory Levels with Predictive Analytics
- Demand Forecasting Techniques
- Case Study: Walmart's Inventory Management System

Fraud Detection and Prevention
- Techniques for Detecting and Preventing Fraudulent Activities
- Real-Time Fraud Detection Systems
- Case Study: PayPal's Fraud Detection Algorithm

Customer Retention and Churn Analysis
- Analyzing Customer Churn and Retention Strategies
- Implementing Loyalty Programs Based on Data Insights
- Case Study: Customer Retention Strategies in E-commerce

9.5. Industry-Specific Case Studies

Big Data in Manufacturing
- Predictive Maintenance and Quality Control
- Supply Chain Optimization
- Case Study: GE's Big Data Strategy for Predictive Maintenance

Big Data in Energy and Utilities
- Smart Grids and Energy Management
- Predictive Analytics for Equipment Maintenance
- Case Study: Big Data for Energy Consumption Optimization

Big Data in Transportation and Logistics
- Route Optimization and Fleet Management
- Real-Time Traffic Analysis and Prediction
- Case Study: DHL's Logistics Optimization with Big Data

Big Data in Telecommunications
- Network Performance Monitoring and Optimization
- Customer Experience and Churn Management
- Case Study: Telecoms Using Big Data for Customer Retention

Big Data in Retail
- Personalized Shopping Experiences and Targeted Promotions
- Inventory and Supply Chain Management
- Case Study: Big Data Analytics in a Major Retail Chain

TEXT BOOKS:

1. Big Data Analytics, Seema Acharya, Subhashini Chellappan, Wiley 2015.
2. R programming for beginners, sandhya arora, latesh malik, university press.

REFERENCE BOOKS:

1. chandramouli subramanian, Asha A Geroge, C R Rene Robin, big data analytics, University press.
2. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Business, Michael Minelli, Michehe Chambers, 1st Edition, Ambiga Dhiraj, Wiley CIO Series, 2013.
3. Hadoop: The Definitive Guide, Tom White, 3rd Edition, O'Reilly Media, 2012.
4. Big Data Analytics: Disruptive Technologies for Changing the Game, Arvind Sathi, 1st Edition, IBM Corporation, 2012.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 702: WEB AND SOCIAL MEDIA ANALYTICS

B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Exposure to various web and social media analytic techniques.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Knowledge on decision support systems
- Apply natural language processing concepts on text analytics
- Understand sentiment analysis
- Knowledge on search engine optimization and web analytics

Course Contents

UNIT-I	**An Overview of Business Intelligence, Analytics, and Decision Support**	**07 Hours**
Analytics to Manage a Vaccine Supply Chain Effectively and Safely, Changing Business Environments and Computerized Decision Support, Information Systems Support for Decision Making, The Concept of Decision Support Systems (DSS), Business Analytics Overview, Brief Introduction to Big Data Analytics		
UNIT-II	**Text Analytics and Text Mining**	**08 Hours**
Machine Versus Men on Jeopardy: The Story of Watson, Text Analytics and Text Mining Concepts and Definitions, Natural Language Processing, Text Mining Applications, Text Mining Process, Text Mining Tools		
UNIT-III	**Sentiment Analysis**	**08 Hours**
Sentiment Analysis Overview, Sentiment Analysis Applications, Sentiment Analysis Process, Sentiment Analysis and Speech Analytics		
UNIT-IV	**Web Analytics, Web Mining**	**07 Hours**
Security First Insurance Deepens Connection with Policyholders, Web Mining Overview, Web Content and Web Structure Mining, Search Engines, Search Engine Optimization, Web Usage Mining (Web Analytics), Web Analytics Maturity Model and Web Analytics Tools		
UNIT-V	**Social Analytics and Social Network Analysis**	**08 Hours**
Social Analytics and Social Network Analysis, Social Media Definitions and Concepts, Social Media Analytics **Prescriptive Analytics - Optimization and Multi-Criteria Systems:** Multiple Goals, Sensitivity Analysis, What-If Analysis, and Goal Seeking		

TEXT BOOK:

1. Ramesh Sharda, Dursun Delen, Efraim Turban, Business Intelligence and Analytics: Systems for Decision Support, Pearson Education

REFERENCE BOOKS:

1. Rajiv Sabherwal, Irma Becerra- Fernandez," Business Intelligence– Practice, Technologies and Management", John Wiley 2011.
2. Lariss T. Moss, Shaku Atre, "Business Intelligence Roadmap", Addison-Wesley It Service.
3. Yuli Vasiliev, "Oracle Business Intelligence: The Condensed Guide to Analysis and Reporting", SPD Shroff, 2012.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2

Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 701L: BIG DATA ANALYTICS LAB
B.Tech. III Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Provide knowledge of Big data Analytics principles and techniques.
- Designed to give an exposure of the frontiers of Big data Analytics

Learning Course Outcome:
After successful completion of the course, students will able to:

- Use Excel as an Analytical tool and visualization tool.
- Ability to program using HADOOP and Map reduce
- Ability to perform data analytics using ML in R.
- Use MongoDB to perform data analytics

Course Contents

List of Experiments	07 Hours

Introduction to Big Data
Lab 1. Exploring Big Data Sets
- Objective: Introduction to large datasets.
- Tools: Python, Pandas, Jupyter Notebook.
- Description: Load and explore a large dataset, summarize its key characteristics, and visualize basic statistics.

Big Data Technologies
Lab 2. Setting Up Hadoop Environment
- Objective: Install and configure Hadoop.
- Tools: Hadoop.
- Description: Set up a single-node Hadoop cluster and explore its basic functionalities.

Lab 3. Working with HDFS
- Objective: Understand HDFS.
- Tools: Hadoop HDFS.
- Description: Perform file operations on HDFS (upload, download, delete files).

Lab 4. MapReduce Programming
- Objective: Write a simple MapReduce program.
- Tools: Hadoop MapReduce, Java.
- Description: Implement a word count example using MapReduce.

Lab 5. Introduction to Spark
- Objective: Get started with Apache Spark.
- Tools: Apache Spark, PySpark.
- Description: Perform basic operations and transformations using Spark.

Lab 6. Comparison of Hadoop and Spark
- Objective: Compare performance between Hadoop and Spark.
- Tools: Hadoop, Spark.
- Description: Implement a simple data processing task and compare execution times.

Data Collection and Storage

Lab 7. Data Collection with Apache Flume
- Objective: Use Flume to collect data.
- Tools: Apache Flume.
- Description: Set up a Flume agent to collect and transport log data.

Lab 8. Data Ingestion with Sqoop
- Objective: Import data from SQL to Hadoop.
- Tools: Apache Sqoop.
- Description: Use Sqoop to import data from MySQL to HDFS.

Lab 9. Setting Up NoSQL Database (MongoDB)
- Objective: Install and use a NoSQL database.
- Tools: MongoDB.
- Description: Install MongoDB, create collections, and perform CRUD operations.

Lab 10. Working with Key-Value Stores
- Objective: Understand key-value stores.
- Tools: Redis.
- Description: Set up Redis, store and retrieve key-value pairs.

Data Processing and Management

Lab 11. Data Cleaning with Pandas
- Objective: Clean and preprocess data.
- Tools: Python, Pandas.
- Description: Perform data cleaning operations such as handling missing values and data normalization.

Lab 12. Data Transformation with Spark
- Objective: Transform data using Spark.
- Tools: Apache Spark, PySpark.
- Description: Perform various data transformation tasks using Spark.

Lab 13. Workflow Orchestration with Apache Airflow
- Objective: Automate data workflows.
- Tools: Apache Airflow.
- Description: Create and schedule a data processing pipeline using Airflow.

Lab 14. Batch Processing with Hadoop
- Objective: Implement batch processing.
- Tools: Hadoop, MapReduce.
- Description: Perform a batch processing task using MapReduce.

Lab 15. Real-Time Processing with Kafka
- Objective: Implement real-time data processing.
- Tools: Apache Kafka, Apache Storm.
- Description: Set up Kafka and Storm to process real-time data streams.

Query Languages and Tools

Lab 16. Querying Big Data with Hive
- Objective: Use Hive for data queries.
- Tools: Apache Hive.

- Description: Write and execute HiveQL queries.

Lab 17. Data Processing with Pig
- Objective: Use Pig for data processing.
- Tools: Apache Pig.
- Description: Write Pig scripts to process data.

Lab 18. Data Analysis with JAQL
- Objective: Analyze data with JAQL.
- Tools: JAQL.
- Description: Write JAQL queries for data analysis.

Lab 19. Advanced Query Optimization in Hive
- Objective: Optimize Hive queries.
- Tools: Apache Hive.
- Description: Implement and test various query optimization techniques in Hive.

Advanced Analytical Methods

Lab 20. Machine Learning with Spark MLlib
- Objective: Implement ML algorithms.
- Tools: Apache Spark MLlib.
- Description: Build and evaluate a machine learning model using Spark MLlib.

Lab 21. Scalable Analytics with R
- Objective: Perform analytics using R.
- Tools: R, SparkR.
- Description: Use SparkR for scalable data analysis.

Lab 22. Predictive Analytics with Python
- Objective: Implement predictive models.
- Tools: Python, scikit-learn.
- Description: Build and evaluate predictive models on large datasets.

Lab 23. Case Study Analysis
- Objective: Analyze real-world data.
- Tools: Various.
- Description: Perform a comprehensive analysis of a case study involving advanced analytics.

Data Visualization and Interpretation

Lab 24. Data Visualization with Tableau
- Objective: Create visualizations.
- Tools: Tableau.
- Description: Use Tableau to create and interpret data visualizations.

Lab 25. Data Visualization with D3.js
- Objective: Visualize data using D3.js.
- Tools: D3.js.
- Description: Create interactive data visualizations using D3.js.

Lab 26. Interpreting Data Insights
- Objective: Communicate insights effectively.
- Tools: Various visualization tools.
- Description: Create a report or presentation to communicate data insights.

Stream Processing and Real-Time Analytics

Lab 27. Building Stream Processing Pipelines
- Objective: Implement a stream processing pipeline.

- Tools: Apache Kafka, Apache Flume.
- Description: Build a stream processing pipeline to process real-time data.

TEXT BOOKS:

1. Big Data Analytics, Seema Acharya, Subhashini Chellappan, Wiley 2015.
2. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Business, Michael Minelli, Michehe Chambers, 1st Edition, Ambiga Dhiraj, Wiely CIO Series, 2013.
3. Hadoop: The Definitive Guide, Tom White, 3rd Edition, O‟Reilly Media, 2012.
4. Big Data Analytics: Disruptive Technologies for Changing the Game, Arvind Sathi, 1st Edition, IBM Corporation, 2012.

REFERENCE BOOKS:

1. Big Data and Business Analytics, Jay Liebowitz, Auerbach Publications, CRC press (2013)
2. Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop, Tom Plunkett, Mark Hornick, McGraw-Hill/Osborne Media (2013), Oracle press.
3. Professional Hadoop Solutions, Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, Wiley, ISBN: 9788126551071, 2015.
4. Understanding Big data, Chris Eaton, Dirk deroos et al, McGraw Hill, 2012.
5. Intelligent Data Analysis, Michael Berthold, David J. Hand, Springer, 2007.
6. Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, Bill Franks, 1st Edition, Wiley and SAS Business Series, 2012.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 731PE: INTERNET OF THINGS (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Computer organization, Computer Networks

Course Objective:

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web-based services on IoT devices

Learning Course Outcome:

After successful completion of the course, students will able to:

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.
- Appraise the role of IoT protocols for efficient network communication.
- Identify the applications of IoT in Industry.

Course Contents

UNIT-I	Introduction to Internet of Things	07 Hours

Definition and Characteristics of IoT, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, IoT Levels and Deployment Templates

Domain Specific IoTs – Home automation, Environment, Agriculture, Health and Lifestyle

UNIT-II	IoT and M2M	08 Hours

M2M, Difference between IoT and M2M, SDN and NFV for IoT,

IoT System Management with NETCOZF, YANG- Need for IoT system Management, Simple Network management protocol, Network operator requirements, NETCONF, YANG, IoT Systems Management with NETCONF-YANG

UNIT-III	IoT Systems – Logical design using Python	08 Hours

Introduction to Python — Python Data types & Data structures, Control flow, Functions, Modules, Packaging, File handling, Data/Time operations, Classes, Exception, Python packages of Interest for IoT

UNIT-IV	IoT Physical Devices and Endpoints	07 Hours
Raspberry Pi, Linux on Raspberry Pi, Raspberry Pi Interfaces, Programming Raspberry PI with Python, Other IoT devices. **IoT Physical Servers and Cloud Offerings** – Introduction to Cloud Storage models and communication APIs, WAMP-AutoBahn for IoT, Xively Cloud for IoT, Python web application framework –Designing a RESTful web API		
UNIT-V	**Case studies**	**08 Hours**
Home Automation, Environment-weather monitoring-weather reporting- air pollution monitoring, Agriculture.		

TEXT BOOK:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547.

REFERENCE BOOK:

1. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 732PE: DATA MINING (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites:
1. Database Management System
2. Probability and Statistics

Course Objective:
- Students will become acquainted with both the strengths and limitations of various data mining techniques like Association, Classification, Cluster and Outlier analysis.

Learning Course Outcome:
After successful completion of the course, students will able to:
- Understand the need of data mining and pre-processing techniques.
- Perform market basket analysis using association rule mining.
- Utilize classification techniques for analysis and interpretation of data.
- Identify appropriate clustering and outlier detection techniques to handle complex data.
- Understand the mining of data from web, text and time series data.

Course Contents

UNIT-I	Introduction to Data Mining	07 Hours

What Data mining? Kinds of Data, Knowledge Discovery process, Data Mining Functionalities, Kinds of Patterns, Major Issues in Data Mining. Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity, Data Pre-processing**:** Major Tasks in Data Pre-processing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT-II	Association Analysis	08 Hours

Basic Concepts, Market Basket Analysis, Apriori Algorithm, FP-growth, From Association Analysis to Correlation Analysis, Pattern Mining in Multilevel Associations and Multidimensional Associations.

UNIT-III	Classification	08 Hours

Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Metrics for Evaluating Classifier Performance, Ensemble Methods, Multilayer Feed- Forward Neural Network, Support Vector Machines, k-Nearest-Neighbor Classifiers.

UNIT- IV	Overview of Basic Clustering Methods	07 Hours

Requirements for Cluster Analysis, Overview of Basic Clustering Methods, Partitioning Methods-k-Means, k-Medoids, Hierarchical Methods-AGENES, DIANA, BIRCH, Density- Based Method-DBSCAN, Outlier Analysis: Types of Outliers, Challenges of Outlier Detection, and Overview of Outlier Detection Methods		
UNIT- V	**Advanced Concepts**	**08 Hours**
Web Mining- Web Content Mining, Web Structure Mining, Web Usage Mining, Spatial Mining- Spatial Data Overview, Spatial Data Mining Primitives, Spatial Rules, Spatial Classification Algorithm, Spatial Clustering Algorithms, Temporal Mining- Modeling Temporal Events, Time Series, Pattern Detection, Sequences, Temporal Association Rules.		

TEXT BOOKS:

1. Jiawei Han, Micheline Kamber, Jian Pei., Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann/Elsevier, 2012.
2. Margaret H Dunham, Data Mining Introductory and Advanced Topics, 2nd Edition, Pearson Education, India, 2006.

REFERENCE BOOKS:

1. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
2. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne and Vipin Kumar, Introduction to Data Mining, 2nd Edition, Pearson Education India, 2021.
3. Amitesh Sinha, Data Warehousing, Thomson Learning, India, 2007.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 733PE: SCRIPTING LANGUAGES (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites:

1. A course on "Computer Programming and Data Structures".
2. A course on "Object Oriented Programming Concepts".

Course Objective:

- This course introduces the script programming paradigm
- Introduces scripting languages such as Perl, Ruby and TCL.
- Learning TCL

Learning Course Outcome:
After successful completion of the course, students will able to:

- Comprehend the differences between typical scripting languages and typical system and application programming languages.
- Gain knowledge of the strengths and weakness of Perl, TCL and Ruby; and select an appropriate language for solving a given problem.
- Acquire programming skills in scripting language

Course Contents

UNIT-I	Introduction	07 Hours
Ruby, Rails, The structure and Execution of Ruby Programs, Package Management with RUBYGEMS, Ruby and web: Writing CGI scripts, cookies, Choice of Webservers, SOAP and web services RubyTk – Simple Tk Application, widgets, Binding events, Canvas, scrolling		
UNIT-II	**Extending Ruby**	**08 Hours**
Ruby Objects in C, the Jukebox extension, Memory allocation, Ruby Type System, Embedding Ruby to Other Languages, Embedding a Ruby Interpreter		
UNIT-III	**Introduction to PERL and Scripting**	**08 Hours**
Scripts and Programs, Origin of Scripting, Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.		

UNIT-IV	Advanced Perl	07 Hours
Finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.		
UNIT-V	TCL/ **Tk**	**08 Hours**
TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures, strings, patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface. **Tk** Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, Perl-Tk.		

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pramatic Progammers guide by Dabve Thomas Second edition

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J.Lee and B. Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E. Quigley, Pearson Education.
3. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.

Perl Power, J. P. Flynt, Cengage Learning.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 734PE: MOBILE APPLICATION DEVELOPMENT
(Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites:

1. Acquaintance with JAVA programming
2. A Course on DBMS

Course Objective:

- To demonstrate their understanding of the fundamentals of Android operating systems
- To improves their skills of using Android software development tools
- To demonstrate their ability to develop software with reasonable complexity on mobile platform
- To demonstrate their ability to deploy software to mobile devices
- To demonstrate their ability to debug programs running on mobile devices

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand the working of Android OS Practically.
- Develop Android user interfaces
- Develop, deploy and maintain the Android Applications.

Course Contents

UNIT-I	Introduction to Android Operating System	07 Hours

Android OS design and Features – Android development framework, SDK features, Installing and running applications on Android Studio, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes
Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

UNIT-II	Android User Interface: Measurements	08 Hours

Device and pixel density independent measuring unit - s Layouts – Linear, Relative, Grid and Table Layouts
User Interface (UI) Components –Editable and non-editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers
Event Handling – Handling clicks or changes of various UI components
Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT-III	**Intents and Broadcasts**	**08 Hours**

Intent — Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS
Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity
Notifications – Creating and Displaying notifications, Displaying Toasts

UNIT-IV	**Persistent Storage**	**07 Hours**

Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences — Creating shared preferences, saving and retrieving data using Shared Preference

UNIT-V	**Database**	**08 Hours**

Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and etindelg data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

TEXT BOOK:

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox), 2012

REFERENCE BOOKS:

1. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013
2. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 741PE: QUANTUM COMPUTING (Professional Elective – IV)
B.Tech. IV Year I Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- To introduce the fundamentals of quantum computing
- The problem-solving approach using finite dimensional mathematics

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand basics of quantum computing
- Understand physical implementation of Qubit
- Understand Quantum algorithms and their implementation
- Understand The Impact of Quantum Computing on Cryptography

Course Contents

UNIT-I	History of Quantum Computing	07 Hours

Importance of Mathematics, Physics and Biology. Introduction to Quantum Computing: Bits Vs Qubits, Classical Vs Quantum logical operations

UNIT-II	Background Mathematics	08 Hours

Basics of Linear Algebra, Hilbert space, Probabilities and measurements. Background Physics: Paul's exclusion Principle, Superposition, Entanglement and super-symmetry, density operators and correlation, basics of quantum mechanics, Measurements in bases other than computational basis. Background Biology: Basic concepts of Genomics and Proteomics (Central Dogma)

UNIT-III	Qubit	08 Hours

Physical implementations of Qubit. Qubit as a quantum unit of information. The Bloch sphere Quantum Circuits: single qubit gates, multiple qubit gates, designing the quantum circuits. Bell states.

UNIT-IV	Quantum Algorithms	07 Hours

Classical computation on quantum computers. Relationship between quantum and classical complexity classes. Deutsch’s algorithm, Deutsch’s-Jozsa algorithm, Shor’s factorization

algorithm, Grover's search algorithm.		
UNIT-V	**Noise and error correction**	**08 Hours**
Graph states and codes, Quantum error correction, fault-tolerant computation. Quantum Information and Cryptography: Comparison between classical and quantum information theory. Quantum Cryptography, Quantum teleportation		

TEXT BOOK:

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge.

REFERENCE BOOKS:

1. Quantum Computing for Computer Scientists by Noson S. Yanofsky and Mirco A. Mannucci
2. Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II.
3. Basic Tools and Special Topics, World Scientific. Pittenger A. O., An Introduction to Quantum Computing Algorithms.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 742PE: EXPERT SYSTEMS (Professional Elective – IV)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Understand the basic techniques of artificial intelligence.
- Understand the Non-monotonic reasoning and statistical reasoning

Learning Course Outcome:
After successful completion of the course, students will able to:

- Apply the basic techniques of artificial intelligence.
- Discuss the architecture of an expert system and its tools.
- Understand the importance of building an expert systems
- Understand various problems with an expert systems

Course Contents

UNIT-I	Introduction to AI programming languages	07 Hours

Blind search strategies, Breadth-first — Depth-first — Heuristic search techniques Hill Climbing – Best first – A Algorithms AO* algorithm – game trees, Min- max algorithms, game playing – Alpha-beta pruning.

UNIT-II	Knowledge representation issues predicate logic	08 Hours

logic programming Semantic nets- frames and inheritance, constraint propagation; Representing Knowledge using rules, Rules-based deduction systems.

UNIT-III	Introduction to Expert Systems	08 Hours

Architecture of expert systems, Representation and organization of knowledge, Basics characteristics, and types of problems handled by expert systems.

UNIT-IV	Expert System Tools	07 Hours

Techniques of knowledge representations in expert systems, knowledge engineering, system-building aids, support facilities, stages in the development of expert systems.

UNIT-V	**Expert System**	**08 Hours**

Building an Expert System: Expert system development, Selection of the tool, Acquiring Knowledge, Building process.

Problems with Expert Systems: Difficulties, common pitfalls in planning, dealing with domain experts, difficulties during development.

TEXT BOOKS:

1. Elain Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw-Hill, New Delhi,
2. Waterman D.A., "A Guide to Expert Systems", Addison Wesley Longman,

REFERENCE BOOKS:

1. Stuart Russel and other Peter Norvig, "Artificial Intelligence – A Modern Approach", Prentice- Hall.
2. Patrick Henry Winston, "Artificial Intelligence", Addison Wesley.
3. Patterson, Artificial Intelligence & Expert System, Prentice Hall India, 1999.
4. Hayes-Roth, Lenat, and Waterman: Building Expert Systems, Addison Wesley, Weiss S. M. and Kulikowski C.A., "A Practical Guide to Designing Expert Systems", Rowman & Allanheld, New Jersey.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO’s-PO’s & PSO’s MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 743PE: CLOUD COMPUTING (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites:

1. A course on "Computer Networks".
2. A course on "Operating System".

Course Objective:

- This course provides an insight into cloud computing
- Topics covered include- Cloud Computing Architecture, Deployment Models, Service Models, Technological Drivers for Cloud Computing, Networking for Cloud Computing and Security in Cloud Computing

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand different computing paradigms and potential of the paradigms and specifically cloud computing
- Understand cloud service types, cloud deployment models and technologies supporting and driving the cloud
- Acquire the knowledge of programming models for cloud and development of software application that runs the cloud and various services available from major cloud providers
- Understand the security concerns and issues in cloud computing
- Acquire the knowledge of advances in cloud computing.

Course Contents

UNIT-I	Cloud Computing Fundamentals	07 Hours

Computing Paradigms, Cloud Computing Fundamentals, Cloud Computing Architecture and Management

UNIT-II	Models	08 Hours

Cloud Deployment Models, Cloud Service Models, Technological Drivers for Cloud Computing: SOA and Cloud, Multicore Technology, Web 2.0 and Web 3.0, Pervasive Computing, Operating System, Application Environment

UNIT-III	Virtualization	08 Hours

Programming Models for Cloud Computing: MapReduce, Cloud Haskell, Software Development in Cloud		
UNIT-IV	**Networking for Cloud Computing**	**07 Hours**
Introduction, Overview of Data Center Environment, Networking Issues in Data Centers, Transport Layer Issues in DCNs, Cloud Service Providers		
UNIT-V	**Security in Cloud Computing**	**08 Hours**
Security in Cloud Computing, and Advanced Concepts in Cloud Computing		

TEXT BOOK:

1. Chandrasekaran, K. Essentials of cloud computing. CRC Press, 2014.

REFERENCE BOOKS:

1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011
2. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, ‘-' - no correlation

BTDS 744PE: GAME THEORY (Professional Elective – IV)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

The course will explain in depth the standard equilibrium concepts (such as Nash equilibrium, Subgame-Perfect Nash Equilibrium, and others) in Game Theory.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand the basic concepts of game theory and solutions
- Understand different types of equilibrium interpretations
- Understand and analyze knowledge and solution concepts
- Analyze extensive games with perfect information

Course Contents

UNIT-I	Introduction	07 Hours

Game Theory, Games and Solutions, Game Theory and the Theory of Competitive Equilibrium, Rational Behavior, The Steady State and Deductive Interpretations, Bounded Rationality Terminology and Notation

Nash Equilibrium- Strategic Games, Nash Equilibrium, Examples, Existence of a Nash Equilibrium, Strictly Competitive Games, Bayesian Games: Strategic Games with Imperfect Information

UNIT-II	Features	08 Hours

Mixed, Correlated, and Evolutionary Equilibrium -Mixed Strategy Nash Equilibrium, Interpretations of Mixed Strategy Nash Equilibrium, Correlated Equilibrium, Evolutionary Equilibrium

Rationalizability and Iterated Elimination of Dominated Actions- Rationalizability Iterated Elimination of Strictly Dominated Actions, Iterated Elimination of Weakly Dominated Actions

UNIT-III	Knowledge and Equilibrium	08 Hours

A Model of Knowledge Common Knowledge, Can People Agree to Disagree? Knowledge and Solution Concepts, The Electronic Mail Game

UNIT-IV	Extensive Games with Perfect Information	07 Hours
Extensive Games with Perfect Information, Subgame Perfect Equilibrium, Two Extensions of the Definition of a Game, The Interpretation of a Strategy, Two Notable Finite Horizon Games, Iterated Elimination of Weakly Dominated Strategies Bargaining Games -Bargaining and Game Theory, A Bargaining Game of Alternating Offers, Subgame Perfect Equilibrium, Variations and Extensions		
UNIT-V	**Repeated Games**	**08 Hours**
The Basic Idea Infinitely Repeated Games vs. Finitely Repeated Games, Infinitely Repeated Games: Definitions, Strategies as Machines, Trigger Strategies: Nash Folk Theorems, Punishing for a Limited Length of Time: A Perfect Folk Theorem for the Limit of Means Criterion, Punishing the Punisher: A Perfect Folk Theorem for the Overtaking Criterion, Rewarding Players Who Punish: A Perfect Folk Theorem for the Discounting Criterion, The Structure of Subgame Perfect Equilibria Under the Discounting Criterion, Finitely Repeated Game		

TEXT BOOKS:

1. A course in Game Theory, M. J. Osborne and A. Rubinstein, MIT Press.

REFERENCE BOOKS:

1. Game Theory, Roger Myerson, Harvard University Press.
2. Game Theory, D. Fudenberg and J. Tirole, MIT Press.
3. Theory of Games and Economic Behavior, J. von Neumann and O. Morgenstern, New York: John Wiley and Sons.
4. Games and Decisions, R.D. Luce and H. Raiffa, New York: John Wiley and Sons.
5. Game Theory, G. Owen, 2nd Edition, New York: Academic Press.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 745PE: KNOWLEDGE REPRESENTATION AND REASONING
(Professional Elective – IV)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Verbal Communication

Course Objective:

- To investigate the key concepts of Knowledge Representation (KR) techniques and different notations.
- To integrate the KR view as a knowledge engineering approach to model organizational knowledge.
- To introduce the study of ontologies as a KR paradigm and applications of ontologies.
- To understand various KR techniques and process, knowledge acquisition and sharing of ontology.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Analyze and design knowledge-based systems intended for computer implementation.
- Acquire theoretical knowledge about principles for logic-based representation and reasoning.
- Ability to understand knowledge-engineering process
- Ability to implement production systems, frames, inheritance systems and approaches to handle uncertain or incomplete knowledge.

Course Contents

UNIT-I	The Key Concepts	07 Hours

Knowledge, Representation, Reasoning, Why knowledge representation and reasoning, Role of logic

Logic: Historical background, Representing knowledge in logic, Varieties of logic, Name, Type, Measures, Unity Amidst diversity

UNIT-II	Ontology	08 Hours

Ontological categories, Philosophical background, Top-level categories, Describing physical entities, Defining abstractions, Sets, Collections, Types and Categories, Space and Time

UNIT-III	Knowledge Representations	08 Hours

Knowledge Engineering, Representing structure in frames, Rules and data, Object-oriented systems, Natural language Semantics, Levels of representation

UNIT-IV	Processes	07 Hours

Times, Events and Situations, Classification of processes, Procedures, Processes and Histories, Concurrent processes, Computation, Constraint satisfaction, Change Contexts: Syntax of contexts, Semantics of contexts, First-order reasoning in contexts, Modal reasoning in contexts, Encapsulating objects in contexts.

UNIT-V	Knowledge Soup	08 Hours

Vagueness, Uncertainty, Randomness and Ignorance, Limitations of logic, Fuzzy logic, Nonmonotonic Logic, Theories, Models and the world, Semiotics Knowledge Acquisition and Sharing: Sharing Ontologies, Conceptual schema, Accommodating multiple paradigms, Relating different knowledge representations, Language patterns, Tools for knowledge acquisition

TEXT BOOKS:

1. Knowledge Representation logical, Philosophical, and Computational Foundations by John F. Sowa, Thomson Learning.
2. Knowledge Representation and Reasoning by Ronald J. Brachman, Hector J. Levesque, Elsevier.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing

skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 721OE: INTRODUCTION TO NATURAL LANGUAGE PROCESSING
(Open Elective – II)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Data structures and compiler design

Course Objective:

- Introduction to some of the problems and solutions of NLP and their relation to linguistics and statistics.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Show sensitivity to linguistic phenomena and an ability to model them with formal grammars.
- Understand and carry out proper experimental methodology for training and evaluating empirical NLP systems
- Able to manipulate probabilities, construct statistical models over strings and trees, and estimate parameters using supervised and unsupervised training methods.
- Able to design, implement, and analyze NLP algorithms; and design different language modeling Techniques.

Course Contents

UNIT-I	Finding the Structure of Words	07 Hours

Words and Their Components, Issues and Challenges, Morphological Models
Finding the Structure of Documents: Introduction, Methods, Complexity of the Approaches, Performances of the Approaches, Features

UNIT-II	Syntax I	08 Hours

Parsing Natural Language, Treebanks: A Data-Driven Approach to Syntax, Representation of Syntactic Structure, Parsing Algorithms.

UNIT-III	Syntax II	08 Hours
Models for Ambiguity Resolution in Parsing, Multilingual Issues **Semantic Parsing I:** Introduction, Semantic Interpretation, System Paradigms, Word Sense.		
UNIT-IV	**Semantic Parsing II**	**07 Hours**
Predicate-Argument Structure, Meaning Representation Systems.		
UNIT-V	**Language Modeling**	**08 Hours**
Introduction, N-Gram Models, Language Model Evaluation, Bayesian parameter estimation, Language Model Adaptation, Language Models- class based, variable length, Bayesian topic based, Multilingual and Cross Lingual Language Modeling		

TEXT BOOKS:

1. Multilingual natural Language Processing Applications: From Theory to Practice – Daniel M. Bikel and Imed Zitouni, Pearson Publication

REFERENCE BOOK:

1. Speech and Natural Language Processing - Daniel Jurafsky& James H Martin, Pearson Publications.
2. Natural Language Processing and Information Retrieval: Tanvier Siddiqui, U.S. Tiwary.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 722OE: AI APPLICATIONS (Open Elective – II)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Fundamentals of AI

Course Objective:

- To give deep knowledge of AI and how AI can be applied in various fields to make life easy.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Correlate AI and solutions to modern problems.
- Use of AI in business applications
- Application of AI in manufacturing automation
- Use of AI in streaming of data and Network applications

Course Contents

UNIT-I	Alibaba	07 Hours
Using Artificial Intelligence To Power The Retail And Business-To-Business Services Of The Future Amazon: Using Deep Learning To Drive Business Performance.		
UNIT-II	**McDonald's**	**08 Hours**
Using Robots And Artificial Intelligence To Automate Processes Walmart: Using Artificial Intelligence To Keep Shelves Stacked And Customers Happy		
UNIT-III	**LinkedIn**	**08 Hours**
Using Artificial Intelligence To Solve The Skills Crisis Netflix: Using Artificial Intelligence To Give Us A Better TV Experience		
UNIT-IV	**Salesforce**	**07 Hours**
How Artificial Intelligence Helps Businesses Understand Their Customers Uber: Using Artificial Intelligence To Do Everything		
UNIT-V	**Siemens**	**08 Hours**

Using Artificial Intelligence And Analytics To Build The Internet Of Trains
Tesla: Using Artificial Intelligence To Build Intelligent Cars

TEXT BOOK:

1. Artificial Intelligence in Practice: How 50 Successful Companies Used AI and Machine Learning to Solve Problems, Bernard Marr, Matt Ward, Wiley.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 711PE: INTERNET of THINGS LAB (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Web Browser Basic knowledge

Course Objective:

- To introduce the raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of distance sensor on IoT devices.

Learning Course Outcome:
After successful completion of the course, students will able to:

1. Ability to introduce the concept of M2M (machine to machine) with necessary protocols and get awareness in implementation of distance sensor
2. Get the skill to program using python scripting language which is used in many IoT devices.

Course Contents

UNIT-I	List of Experiments	07 Hours

1. Using Raspberry pi
   a. Calculate the distance using a distance sensor.
   b. Interface an LED and switch with Raspberry pi.
   c. Interface an LDR with Raspberrry Pi.

2. Using Arduino
   a. Calculate the distance using a distance sensor.
   b. Interface an LED and switch with Aurdino.
   c. Interface an LDR with Aurdino
   d. Calculate temperature using a temperature sensor.

3. Using Node MCU
   a. Calculate the distance using a distance sensor.
   b. Interface an LED and switch with Raspberry pi.
   c. Interface an LDR with Node MCU
   d. Calculate temperature using a temperature sensor.

4. Installing OS on Raspberry Pi
   a) Installation using PiImager

b) Installation using image file

- Downloading an Image
- Writing the image to an SD card
  - using Linux
  - using Windows
- Booting up Follow the instructions given in the URL https://www.raspberrypi.com/documentation/computers/getting-started.html

5. Accessing GPIO pins using Python

a) Installing GPIO Zero library.

update your repositories list:
install the package for Python 3:

b) Blinking an LED connected to one of the GPIO pin

c) Adjusting the brightness of an LED Adjust the brightness of an LED (0 to 100, where 100 means maximum brightness) using the in-built PWM wavelength.

6. Create a DJANGO project and an app.
7. Create a DJANGO view for weather station REST API
8. Create DJANGO template
9. Configure MYSQL with DJANGO framework

TEXT BOOKS:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547.
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759.

REFERENCE BOOKS:

1. Bernd Scholz-Reiter, Florian Michahelles, “Architecting the Internet of Things”, ISBN 978-3- 642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016
2. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students’ reading and writing

skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 712PE: DATA MINING LAB (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

- **Course Prerequisites: A course on "Database Management System**

Course Objective:

- The course is intended to obtain hands-on experience using data mining software.
- Intended to provide practical exposure of the concepts in data mining algorithms

Learning Course Outcome:
After successful completion of the course, students will able to:

1. Apply preprocessing statistical methods for any given raw data.
2. Gain practical experience of constructing a data warehouse.
3. Implement various algorithms for data mining in order to discover interesting patterns from large amounts of data.
4. Apply OLAP operations on data cube construction.

Course Contents

UNIT-I	LIST OF EXPERIMENTS	07 Hours

Experiments using Weka/ Pentaho/Python

1. Data Processing Techniques:
   (i) Data cleaning (ii) Data transformation – Normalization (iii) Data integration
2. Partitioning - Horizontal, Vertical, Round Robin, Hash based
3. Data Warehouse schemas – star, snowflake, fact constellation
4. Data cube construction – OLAP operations
5. Data Extraction, Transformations & Loading operations
6. Implementation of Attribute oriented induction algorithm
7. Implementation of apriori algorithm
8. Implementation of FP – Growth algorithm
9. Implementation of Decision Tree Induction
10. Calculating Information gain measures
11. Classification of data using Bayesian approach
12. Classification of data using K – nearest neighbour approach
13. Implementation of K – means algorithm
14. Implementation of BIRCH algorithm

15. Implementation of PAM algorithm
16. Implementation of DBSCAN algorithm

TEXT BOOKS:

1. Data Mining – Concepts and Techniques - JIAWEI HAN &MICHELINE KAMBER, Elsevier.
2. Data Warehousing, Data Mining &OLAP- Alex Berson and Stephen J. Smith- Tata McGraw-Hill Edition, Tenth reprint 2007

REFERENCE BOOK:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Anuj Karpatne, Introduction to Data Mining, Pearson Education.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 713PE: SCRIPTING LANGUAGES LAB (Professional Elective – III)
B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Any High level programming language (C, C++)

Course Objective:

- To Understand the concepts of scripting languages for developing web based projects
- To understand the applications the of Ruby, TCL, Perl scripting languages

Learning Course Outcome:
After successful completion of the course, students will able to:

1. Ability to understand the differences between Scripting languages and programming languages
2. Gain some fluency programming in Ruby, Perl, TCL.

Course Contents

UNIT-I	LIST OF EXPERIMENTS	07 Hours

1. Write a Ruby script to create a new string which is n copies of a given string where n is a non-negative integer
2. Write a Ruby script which accept the radius of a circle from the user and compute the parameter and area.
3. Write a Ruby script which accept the users first and last name and print them in reverse order with a space between them
4. Write a Ruby script to accept a filename from the user print the extension of that
5. Write a Ruby script to find the greatest of three numbers
6. Write a Ruby script to print odd numbers from 10 to 1
7. Write a Ruby script to check two integers and return true if one of them is 20 otherwise return their sum
8. Write a Ruby script to check two temperatures and return true if one is less than 0 and the other is greater than 100
9. Write a Ruby script to print the elements of a given array
10. Write a Ruby program to retrieve the total marks where subject name and marks of a student stored in a hash
11. Write a TCL script to find the factorial of a number
12. Write a TCL script that multiplies the numbers from 1 to 10

13. Write a TCL script for sorting a list using a comparison function
14. Write a TCL script to (i) create a list (ii) append elements to the list (iii) Traverse the list (iv) Concatenate the list
15. Write a TCL script to comparing the file modified times.
16. Write a TCL script to Copy a file and translate to native format.
17. a) Write a Perl script to find the largest number among three numbers.
    b) Write a Perl script to print the multiplication tables from 1-10 using subroutines.
18. Write a Perl program to implement the following list of manipulating functions
    a) Shift
    b) Unshift
    c) Push
19. a) Write a Perl script to substitute a word, with another word in a string.
    b) Write a Perl script to validate IP address and email address.
20. Write a Perl script to print the file in reverse order using command line arguments

TEXT BOOKS:

1. The World of Scripting Languages, David Barron,Wiley Publications.
2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pramatic Progammers guide by Dabve Thomas Second edition

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J.Lee and B. Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E. Quigley, Pearson Education.
3. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
5. Perl Power, J. P. Flynt, Cengage Learning.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 714PE: MOBILE APPLICATION DEVELOPMENT LAB (Professional Elective – III)

B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- To learn how to develop Applications in an android environment.
- To learn how to develop user interface applications.
- To learn how to develop URL related applications.

Learning Course Outcome:

After successful completion of the course, students will able to:

1. Understand the working of Android OS Practically.
2. Develop user interfaces.
3. Develop, deploy and maintain the Android Applications.

Course Contents

UNIT-I	LIST OF EXPERIMENTS	07 Hours

1. Create an Android application that shows Hello + name of the user and run it on an emulator.
(b) Create an application that takes the name from a text box and shows hello message along with the name entered in the text box, when the user clicks the OK button.
2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Datepicker), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.
3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a “Back” button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on the right fragment instead of the second screen with the back button. Use Fragment transactions and Rotation event listeners.
4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.

5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.
6. Create an application that uses a text file to store usernames and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with a Login Failed message.
7. Create a user registration application that stores the user details in a database table.
8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.
9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.
10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.
11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.
12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.

TEXT BOOKS:

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox), 2012.
2. Android Application Development for Java Programmers, James C Sheusi, Cengage, 2013.

REFERENCE BOOK:

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 715PE: CRYPTOGRAPHY AND NETWORK SECURITY LAB (Professional Elective – III)

B.Tech. IV Year I Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand basic cryptographic algorithms, message and web authentication and security issues.
- Identify information system requirements for both of them such as client and server.
- Understand the current legal issues towards information security.

Course Contents

UNIT-I	List of Experiments	07 Hours

1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and display the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms
   a. Ceaser cipher b. Substitution cipher c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement the RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.

10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA

TEXT BOOKS:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, McGraw Hill, 3rd Edition

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security: Forouzan Mukhopadhyay, McGraw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 851PE: SOCIAL NETWORK ANALYSIS (Professional Elective – V)

B.Tech. IV Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites:
- Web Technologies
- Computer Networks
- Data Warehousing and Data Mining

Course Objective:
- Understand the concepts of social media
- Learn the mechanisms for social network analysis
- Analysis of widely used services such as email, Wikis, Twitter, flickr, YouTube, etc.

Learning Course Outcome:
After successful completion of the course, students will able to:
- Ability to construct social network maps easily
- Gain skills in tracking the content flow through the social media
- Understand NodeXL use to perform social network analysis

Course Contents

UNIT-I	Introduction	07 Hours

Social Media and Social Networks
Social Media: New Technologies of Collaboration
Social Network Analysis: Measuring, Mapping, and Modelling collections of Connections.

UNIT-II	Preparing Data	08 Hours

NodeXL, Layout, Visual Design, and Labelling, Calculating and Visualising Network Metrics, Preparing Data and Filtering, Clustering and Grouping.

UNIT-III	CASE STUDIES	08 Hours

Email: The lifeblood of Modern Communication.
Thread Networks: Mapping Message Boards and Email Lists
Twitter: Conversation, Entertainment and Information

UNIT-IV	CASE STUDIES	07 Hours

Visualizing and Interpreting Facebook Networks, WWW Hyperlink Networks		
UNIT-V	**CASE STUDIES**	**08 Hours**
You Tube: Contrasting Patterns of Content Interaction, and Prominence**.** **Wiki Networks:** Connections of Creativity and Collaboration		

TEXT BOOK:

1. Hansen, Derek, Ben Sheiderman, Marc Smith, Analyzing Social Media Networks with NodeXL: Insights from a Connected World, Morgan Kaufmann, 2011.

REFERENCE BOOKS:

1. Avinash Kaushik, Web Analytics 2.0: The Art of Online Accountability, Sybex, 2009.
2. Marshall Sponder, Social Media Analytics: Effective Tools for Building, Interpreting and Using Metrics, 1st Edition, MGH, 2011.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 852PE: FEDERATED MACHINE LEARNING
(Professional Elective – V)

B.Tech. IV Year II Sem.

Teaching Scheme: **TH: - 4 Hours/Week**	**Credit** **TH: 04**	**Examination Scheme:** **In Sem. Evaluation: 25 Marks** **Mid Sem. Exam: 25 Marks** **End Sem. Exam: 50 Marks** **Total : 100 Marks**

Course Prerequisites:

- The prerequisite knowledge for this course includes machine learning, basic computer systems and basic programming skills.

Course Objective:

- Understand the key concepts and issues behind Federated Learning
- Get familiar with key theoretical results of Federated Learning

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand the basics on privacy-preserving ML
- Analyze the key concepts of Distributed ML and FL
- Understand the key concepts and applications of Horizontal FL and Vertical FL
- Motivates the intensive mechanism design for FL
- Analyze the concepts of federated reinforcement learning

Course Contents

UNIT-I	**Introduction**	**07 Hours**

Motivation, Federated Learning as a Solution, The Definition of Federated Learning, Categories of Federated Learning, Current Development in Federated Learning, Research Issues in Federated Learning, Open-Source Projects, Standardization Efforts, The Federated AI Ecosystem Background: Privacy-Preserving Machine Learning, PPML and Secure ML, Threat and Security Models, Privacy Threat Models, Adversary and Security Models, Privacy Preservation Techniques, Secure Multi-Party Computation, Homomorphic Encryption, Differential Privacy.

UNIT-II	**Distributed Machine Learning**	**08 Hours**

Introduction to DML, The Definition of DML, DML Platforms, Scalability- Motivated DML, Large-Scale Machine Learning, Scalability-Oriented DML Schemes, Privacy-Motivated DML, Privacy-Preserving Decision Trees, Privacy-Preserving Techniques, Privacy-Preserving DML Schemes, Privacy-Preserving Gradient Descent, Vanilla Federated Learning, Privacy-Preserving Methods

UNIT-III	**Horizontal Federated Learning**	**08 Hours**

The Definition of HFL, Architecture of HFL, The Client- Server Architecture, The Peer-to-Peer Architecture, Global Model Evaluation, The Federated Averaging Algorithm, Federated Optimization, The FedAvg Algorithm, The Secured FedAvg Algorithm, Improvement of the FedAvg Algorithm, Communication Efficiency, Client Selection Vertical Federated Learning: The Definition of VFL, Architecture of VFL, Algorithms of VFL, Secure Federated Linear Regression, Secure Federated Tree-Boosting

UNIT-IV	**Federated Transfer Learning**	**07 Hours**

Heterogeneous Federated Learning, Federated Transfer Learning, The FTL Framework, Additively Homomorphic Encryption, The FTL Training Process, The FTL Prediction Process, Security Analysis, Secret Sharing-Based FTL Incentive Mechanism Design for Federated Learning: Paying for Contributions, Profit- Sharing Games, Reverse Auctions, A Fairness-Aware Profit Sharing Framework, Modeling Contribution, Modeling Cost, Modeling Regret, Modeling Temporal Regret, The Policy Orchestrator, Computing Payoff Weightage

UNIT-V	**Federated Learning**	**08 Hours**

for Vision, Language, and Recommendation: Federated Learning for Computer Vision, Federated CV, Federated Learning for NLP, Federated NLP, Federated Learning for Recommendation Systems, Recommendation Model, Federated Recommendation System

Federated Reinforcement Learning:

Introduction to Reinforcement Learning, Policy, Reward, Value Function, Model of the Environment, RL Background Example, Reinforcement Learning Algorithms, Distributed Reinforcement Learning, Asynchronous Distributed Reinforcement Learning, Synchronous Distributed Reinforcement Learning, Federated Reinforcement Learning, Background and Categorization

TEXT BOOK:

1. Federated Learning, Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, and Han Yu Synthesis Lectures on Artificial Intelligence and Machine Learning 2019.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 853PE: AUGMENTED REALITY & VIRTUAL REALITY

(Professional Elective –V) B.Tech. IV Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Provide a foundation to the fast-growing field of AR and make the students aware of the various AR concepts**.**
- To give historical and modern overviews and perspectives on virtual reality. It describes the fundamentals of sensation, perception, technical and engineering aspects of virtual reality systems.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Describe how AR systems work and list the applications of AR.
- Understand the software architectures of AR.
- Understand the Visual perception and rendering in VR
- Understand the interaction, auditory perception and rendering in VR

Course Contents

UNIT-I	Introduction to Augmented Reality	07 Hours

Augmented Reality - Defining augmented reality, history of augmented reality, Examples, Related fields

Displays: Multimodal Displays, Visual Perception, Requirements and Characteristics, Spatial Display Model, Visual Displays

Tracking: Tracking, Calibration, and Registration, Coordinate Systems, Characteristics of Tracking Technology, Stationary Tracking Systems, Mobile Sensors

UNIT-II	Computer Vision for Augmented Reality	08 Hours

Marker Tracking, Multiple-Camera Infrared Tracking, Natural Feature Tracking by Detection, Outdoor Tracking. **Interaction**: Output Modalities, Input Modalities, Tangible Interfaces, Virtual User Interfaces on Real Surfaces, Augmented Paper, Multi-view Interfaces, Haptic Interaction **Software Architectures**: AR Application Requirements, Software Engineering Requirements, Distributed Object Systems, Dataflow, Scene Graphs

UNIT-III	Introduction to Virtual Reality	08 Hours
Defining Virtual Reality, History of VR, Human Physiology and Perception **The Geometry of Virtual Worlds:** Geometric Models, Axis-Angle Representations of Rotation, Viewing Transformations **Light and Optics:** Basic Behavior of Light, Lenses, Optical Aberrations, The Human Eye, Cameras, Displays		
UNIT-IV	**The Physiology of Human Vision**	**07 Hours**
From the Cornea to Photoreceptors, From Photoreceptors to the Visual Cortex, Eye Movements, Implications for VR **Visual Perception:** Visual Perception - Perception of Depth, Perception of Motion, Perception of Color **Visual Rendering:** Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates, Immersive Photos and Videos.		
UNIT-V	**Motion in Real and Virtual Worlds**	**08 Hours**
Velocities and Accelerations, The Vestibular System, Physics in the Virtual World, Mismatched Motion and Vection **Interaction:** Motor Programs and Remapping, Locomotion, Social Interaction **Audio:** The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering		

TEXT BOOKS:

1. Augmented Reality: Principles & Practice by Schmalstieg / Hollerer, Pearson Education India;First edition (12 October 2016),ISBN-10: 9332578494
2. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016

REFERENCE BOOKS:

1. Allan Fowler-AR Game Development‖, 1st Edition, A press Publications, 2018, ISBN 978- 1484236178
2. Understanding Virtual Reality: Interface, Application and Design, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)". Morgan Kaufmann Publishers, San Francisco, CA, 2002
3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009
4. Designing for Mixed Reality, Kharis O'Connell Published by O'Reilly Media, Inc., 2016, ISBN: 9781491962381

5. Sanni Siltanen- Theory and applications of marker-based augmented reality. Julkaisija — Utgivare Publisher. 2012. ISBN 978-951-38-7449-0
6. Gerard Jounghyun Kim, "Designing Virtual Systems: The Structured Approach", 2005.

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2

2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 854PE: WEB SECURITY (Professional Elective –V)

B.Tech. IV Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

1. Give an Overview of information security
2. Give an overview of Access control of relational databases

Learning Course Outcome:
After successful completion of the course, students will able to:

1. Understand the Web architecture and applications
2. Understand client side and service side programming
3. Understand how common mistakes can be bypassed and exploit the application
4. Identify common application vulnerabilities

Course Contents

UNIT-I	The Web Security	07 Hours
The Web Security Problem, Risk Analysis and Best Practices Cryptography and the Web: Cryptography and Web Security, Working Cryptographic Systems and Protocols, Legal Restrictions on Cryptography, Digital Identification.		
UNIT-II	**The Web's War on Your Privacy**	**08 Hours**
Privacy-Protecting Techniques, Backups and Antitheft, Web Server Security, Physical Security for Servers, Host Security for Servers, Securing Web Applications.		
UNIT-III	**Database Security**	**08 Hours**
Recent Advances in Access Control, Access Control Models for XML, Database Issues in Trust Management and Trust Negotiation, Security in Data Warehouses and OLAP Systems		
UNIT-IV	**Security Re-engineering for Databases**	**07 Hours**
Concepts and Techniques, Database Watermarking for Copyright Protection, Trustworthy Records Retention, Damage Quarantine and Recovery in Data Processing Systems, Hippocratic Databases: Current Capabilities.		
UNIT-V	**Future Trends Privacy in Database Publishing**	**08 Hours**

A Bayesian Perspective, Privacy-enhanced Location Based Access Control, Efficiently Enforcing the Security and Privacy Policies in a Mobile Environment.
TEXT BOOKS: 1. Web Security, Privacy and Commerce Simson G Arfinkel, Gene Spafford, O'Reilly. Handbook on Database security applications and trends Michael Gertz, Sushil Jajodia

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 855PE: AD-HOC & SENSOR NETWORKS (Professional Elective – V)
B.Tech. IV Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Computer Networks, Distributed Systems, Mobile Computing

Course Objective:
- To understand the challenges of routing in ad-hoc and sensor networks
- To understand various broadcast, mutlicast and geocasting protocols in ad hoc and sensor networks
- To understand basics of Wireless sensors, and Lower Layer Issues and Upper Layer Issues of WSN

Learning Course Outcome:
After successful completion of the course, students will able to:
- Understand the concepts of sensor networks and applications
- Understand and compare the MAC and routing protocols for adhoc networks
- Understand the transport protocols of sensor networks

Course Contents

UNIT-I	Introduction to Ad Hoc Networks	07 Hours

Characteristics of MANETs, Applications of MANETs and Challenges of MANETs.
Routing in MANETs
Criteria for classification, Taxonomy of MANET routing algorithms, *Topology-based* routing algorithms- Proactive: DSDV, WRP; Reactive: DSR, AODV, TORA; Hybrid: ZRP; *Position-based* routing algorithms- Location Services-DREAM, Quorum-based, GLS; Forwarding Strategies, Greedy Packet, Restricted Directional Flooding-DREAM, LAR; Other routing algorithms-QoS Routing, CEDAR.

UNIT-II	Data Transmission	08 Hours

Broadcast Storm Problem, Rebroadcasting Schemes-Simple-flooding, Probability-based Methods, Area- based Methods, Neighbour Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree-based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AMRoute, MCEDAR.

UNIT-III	Geocasting	08 Hours

Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR.
TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

UNIT-IV	Basics of Wireless Sensors and Lower Layer Issues	07 Hours

Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.

UNIT-V	Upper Layer Issues of WSN	08 Hours

Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

TEXT BOOKS

1. Ad Hoc and Sensor Networks – Theory and Applications, *Carlos Corderio Dharma P.Aggarwal*, World Scientific Publications, March 2006, ISBN — 981-256-681-3
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN — 978-1-55860-914-3 (Morgan Kauffman)

REFERENCE BOOKS:

1. C. Siva Ram Murthy, B.S. Manoj Ad Hoc Wireless Networks: Architectures and Protocols.
2. Taieb Znati Kazem Sohraby, Daniel Minoli, Wireless Sensor Networks: Technology, Protocols and Applications, Wiley.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 861PE: SPEECH AND VIDEO PROCESSING (Professional Elective – VI)
B.Tech. IV Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- To make students understand speech and video processing techniques

Learning Course Outcome:
After successful completion of the course, students will able to:

- Describe the mechanisms of human speech production systems and methods for speech feature extraction.
- Understand basic algorithms of speech analysis and speech recognition.
- Explain basic techniques in digital video processing, including imaging characteristics and sensors.
- Apply motion estimation and object tracking algorithms on video.

Course Contents

UNIT-I	Speech processing concepts	07 Hours
The speech production mechanism, Discrete time speech signals, Pole-Zero modeling of speech, relevant properties of the fast Fourier transform for speech recognition, convolution, linear and nonlinear filter banks, spectral estimation of speech using DFT. Linear Prediction analysis of speech.		
UNIT-II	**Speech recognition**	**08 Hours**
Feature extraction for speech, static and dynamic feature for speech recognition, MFCC, LPCC, Distance measures, vector quantization models, Gaussian Mixture model, HMM.		
UNIT-III	**Multi-Dimensional Signals and Systems**	**08 Hours**

Multi-Dimensional Signals, Multi-Dimensional Transforms, Multi-Dimensional Systems, Multi- Dimensional Sampling Theory, Sampling Structure Conversion
Digital Images and Video: Human Visual System and Color, Digital Video

UNIT-IV	Motion Estimation	07 Hours

Image Formation, Motion Models, 2D Apparent-Motion Estimation, Differential Methods, Matching Methods, Nonlinear Optimization Methods, Transform-Domain Methods, 3D Motion and Structure Estimation

UNIT-V	Video Segmentation and Tracking	08 Hours

Image Segmentation, Change Detection, Motion Segmentation, Motion Tracking, Image and Video Matting, Performance Evaluation

TEXT BOOKS:

1. Fundamentals of Speech recognition – L. Rabiner and B. Juang, Prentice Hall signal processing series
2. Digital Video processing, A Murat Tekalp, 2nd edition, Prentice Hall.

REFERENCE BOOKS:

1. Discrete-time speech signal processing: principles and practice, Thomas F. Quatieri, Coth.
2. Video Processing and Communications, Yao Wang, J. Osternann and Qin Zhang, Pearson Education.
3. “Speech and Audio Signal Processing”, B.Gold and N. Morgan, Wiley.
4. “Digital image sequence processing, Compression, and analysis”, Todd R. Reed, CRC Press.
5. “Handbook of Image and Video processing”, Al Bovik, Academic press, second Edition.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 862PE: ROBOTIC PROCESS AUTOMATION (Professional Elective – VI)
B.Tech. IV Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Introduce robotic process automation, techniques of automation using UIPath RPA tool.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand the concepts of Robotic Process Automation.
- Apply the flow chart mechanism in various calculations.
- Applying UIPath tool for debugging process.
- Design system managing techniques.
- Create application for process automation using UIPath tool.

Course Contents

UNIT-I	Robotic Process Automation	07 Hours

UNIT - I
Introduction, Scope and techniques of automation, Robotic process automation, Components of RPA, RPA platforms, About UiPath
UIPath Stack Uipath Studio, Uipath Robot, Types of Robots, UiPath Orchestrator
UIPath Studio Projects, User interface
The User Interface: Task recorder, Advanced UI interactions: Input methods, Output methods.

UNIT-II	Sequence, Flowchart, and Control Flow	08 Hours

Sequencing the workflow, Activities, Control Flow, various types of loops and decision making
Data Manipulation: Variables and scope, Collections, Arguments – Purpose and use, Data table usage with examples, File operation with step-by-step example, CSV/Excel to data table and vice versa.

UNIT-III	Taking Control of the Controls	08 Hours

Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls — mouse and keyboard activities, Handling events, revisit recorder, When to use OCR, Types of OCR available, How to use OCR

Plugins and Extensions: Terminal Plugin, SAP Automation, Citrix automation and Credential management.

UNIT-IV	Handling User Events and Assistant Bots	07 Hours

Assistant bots, Monitoring system event triggers, Monitoring image and element triggers, Launching an assistant bot on a keyboard event

Exception Handling, Debugging, and Logging: Exception handling, Common exceptions and ways to handle them, Logging and taking screenshots, Debugging techniques, Collecting crash dumps, Error reporting.

UNIT-V	Managing and Maintaining the Code	08 Hours

Project organization, nesting workflows, Reusability of workflows, Commenting techniques, State Machine, When to use Flowcharts, State Machines, or Sequences, Using config files

Deploying and Maintaining the Bot: Publishing using publish utility, using Orchestration Server to control bots, deploy bots, License Management, Publishing and Managing updates

TEXT BOOKS:

1. Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool - UiPath: Create Software robots. with the leading RPA tool — UiPath Kindle Edition

REFERENCE BOOKS:

1. Robotic Process Automation A Complete Guide - 2020 Edition Kindle Edition.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 863PE: RANDOMIZED ALGORITHMS (Professional Elective – VI)
B.Tech. IV Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:
- To introduce the power of randomization in the design of algorithms.

Learning Course Outcome:
After successful completion of the course, students will able to:
- Appreciate the fundamentals of randomized algorithm design.
- Understand the fundamentals of Markov chains and the Monte Carlo method.
- Apply high probability analysis to selected randomized algorithms.
- Understand the Fingerprint and Pattern Matching techniques

Course Contents

UNIT-I	Introduction, A Min	07 Hours

Cut algorithm, Las Vegas and Monte Carlo, Binary Planar Partitions, A Probabilistic Recurrence
Game-Theoretic Techniques: Game Tree Evaluation, The Minimax Principle

UNIT-II	Moments and Deviations	08 Hours

Occupancy Problems, The Markov and Chebyshev Inequalities, Randomized Selection, Two Point sampling, The Coupon Collector's problem.
Markov Chains and Random Walks: A 2-SAT example, Markov Chains, Random Walks on Graphs, Graph Connectivity

UNIT-III	Algebraic Techniques	08 Hours

Fingerprinting and Freivald's Technique, Verifying Polynomial Identities, Perfect Matching in Graphs, Verifying Equality of Strings, A Comparison of Fingerprinting Techniques, Pattern Matching

UNIT-IV	Data Structures	07 Hours

The Fundamental of Data-structures, Random Treaps, Skip Lists, Hash Tables Graph Algorithms: All Pairs Shortest Path, The Min- Cut Problem, Minimum Spanning Trees		
UNIT-V	**Geometric Algorithms**	**08 Hours**
Randomized Incremental Construction, Convex Hulls in the Plane, Duality, Half- Space Intersections, Dalaunay Triangulations, Trapezoidal Decompositions, Parallel and Distributed Algorithms: The PRAM Model, Sorting on a PRAM, Maximal Independent Sets, Perfect Matchings		
TEXT BOOKS: 1. Randomized Algorithms: Rajeev Motwani, Prabhakar Raghavan, cambridge University Press 2. Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis by Eli Upfal and Michael Mitzenmacher.		

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 864PE: COGNITIVE COMPUTING (Professional Elective – VI)
B.Tech. IV Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Probability theory

Course Objective:

- To provide an understanding of the central challenges in realizing aspects of human cognition.
- To provide a basic exposition to the goals and methods of human cognition.
- To develop algorithms that use AI and machine learning along with human interaction and feedback to help humans make choices/decisions.
- To support human reasoning by evaluating data in context and presenting relevant findings along with the evidence that justifies the answers.

Learning Course Outcome:

After successful completion of the course, students will able to:

- Understand cognitive computing
- Plan and use the primary tools associated with cognitive computing.
- Plan and execute a project that leverages cognitive computing.
- Understand and develop the business implications of cognitive computing.

Course Contents

UNIT-I	Introduction to Cognitive Science	07 Hours

Understanding Cognition, IBM's Watson, Design for Human Cognition, Augmented Intelligence, Cognition Modeling Paradigms: Declarative/ logic-based computational cognitive modeling, connectionist models of cognition, Bayesian models of cognition, a dynamical systems approach to cognition.

UNIT-II	Cognitive Models	08 Hours

Cognitive Models of memory and language, computational models of episodic and semantic memory, modeling psycholinguistics. Cognitive Modeling: modeling the interaction of language, memory and learning, Modeling select aspects of cognition classical models of rationality, symbolic reasoning and decision making.

UNIT-III	Formal models	08 Hours
Formal models of inductive generalization, causality, categorization and similarity, the role of analogy in problem solving, Cognitive Development Child concept acquisition. Cognition and Artificial cognitive architectures such as ACT-R, SOAR, OpenCog, CopyCat, Memory Networks.		
UNIT-IV	**Deep QA Architecture**	**07 Hours**
Unstructured Information Management Architecture (UIMA), Structured Knowledge, Business Implications, Building Cognitive Applications, Application of Cognitive Computing and Systems.		

TEXT BOOK:

1. The Cambridge Handbook of Computational Psychology by Ron Sun (ed.), Cambridge University Press.

REFERENCE BOOKS:

1. Judith S. Hurwitz, Marcia Kaufman, Adrian Bowles Cognitive Computing and Big Data Analytics, Wiley
2. Vijay V Raghavan, Venkat N. Gudivada, Venu Govindaraju, Cognitive Computing: Theory and Applications: Volume 35 (Handbook of Statistics), North Hollan.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 865PE: SEMANTIC WEB (Professional Elective – VI)
B.Tech. IV Year II Sem.

Teaching Scheme: TH: - 4 Hours/Week	Credit TH: 04	Examination Scheme: In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Introduce Semantic Web Vision and learn Web intelligence
- Understanding about XML, RDF, RDFS, OWL
- Querying Ontology and Ontology Reasoning
- To learn Semantic Web Applications, Services and Technology
- To learn Knowledge Representation for the Semantic Web

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand the characteristics of the semantic web technology
- Understand the concepts of Web Science, semantics of knowledge resource and ontolog
- Describe logic semantics and inference with OWL.
- Use ontology engineering approaches in semantic applications
- Learn about web graph processing for various applications such as search engine, community detection

Course Contents

UNIT-I	Introduction	07 Hours

Introduction to Semantic Web, the Business Case for the Semantic Web, XML and Its Impact on the Enterprise.

UNIT-II	Web Services	08 Hours

Uses, Basics of Web Services, SOAP, UDDI, Orchestrating Web Services, Securing Web Services, Grid Enabled and Semantic Web of Web Services.

UNIT-III	Resource Description Framework	08 Hours

Features, Capturing Knowledge with RDF.
XML Technologies: XPath, The Style Sheet Family: XSL, XSLT, and XSL FO, XQuery, XLink, XPointer, XInclude, XMLBase, XHTML, XForms, SVG.

UNIT-IV	Taxonomies and Ontologies	07 Hours
Overview of Taxonomies, Defining the Ontology Spectrum, Topic Maps, Overview of Ontologies, Syntax, Structure, Semantics, and Pragmatics, Expressing Ontologies Logically, Knowledge Representation.		
UNIT-V	**Semantic Web Application**	**08 Hours**
Semantic Web Services, e-Learning, Semantic Bioinformatics, Enterprise Application Integration, Knowledge Base. **Semantic Search Technology:** Search Engines, Semantic Search, Semantic Search Technology, Web Search Agents, Semantic Methods, Latent Semantic Index Search, TAP, Swoogle.		

TEXT BOOKS:

1. The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management by Michael C. Daconta, Leo J. Obrst , Kevin T. Smith, Wiley Publishing, Inc.
2. Peter Mika, Social Networks and the Semantic Web, Springer

REFERENCE BOOKS:

1. Thinking on the Web - Berners Lee, Godel and Turing, Wiley Interscience
2. The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management by Michael C. Daconta, Leo J. Obrst , Kevin T. Smith, Wiley Publishing, Inc.
3. Semantic Web Technologies, Trends and Research in Ontology Based Systems, J. Davies, R. Studer, P. Warren, John Wiley & Sons.
4. Semantic Web and Semantic Web Services - Liyang Lu Chapman and Hall/CRC Publishers, (Taylor & Francis Group)
5. Information Sharing on the semantic Web - Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.
6. Programming the Semantic Web, T. Segaran, C. Evans,J. Taylor, O'Reilly, SPD.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 831OE: CHATBOTS (Open Elective – III)
B.Tech. IV Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: None

Course Objective:

- Knowledge on concepts of chatbots and understanding the developer environment bot framework.

Learning Course Outcome:
After successful completion of the course, students will able to:

- Understand basic concepts of chatbots
- Analyze different entities in building bots
- Understand the concepts of advanced bot building
- Discuss different types of chatbot use cases

Course Contents

UNIT-I	Introduction to Chatbots	07 Hours
Definition of chatbots, Journey of Chatbots, Rise of Chatbots, Messaging Platforms		
UNIT-II	**Setting Up the Developer Environment Botframework**	**08 Hours**
Local Installation, Installing NodeJS, Following the Development Pipeline, Storing Messages in Database.		
UNIT-III	**Basics of Bot Building**	**08 Hours**
Basics of Bot Building- Intents, Entities		
UNIT-IV	**Advanced Bot Building**	**07 Hours**
Design Principles, Showing Product Results, Saving Messages, Building Your Own Intent Classifier		
UNIT-V	**Business and Monetization**	**07 Hours**

Analytics, Chatbot Use Cases- Modes of Communication- Business-to-Business (B2B), ChapBusiness- to-Consumer (B2C) Consumer-to-Consumer (C2C) Business-to-Employee (B2E), Employee-to- Employee (E2E), Chatbots by Industry Vertical
TEXT BOOK: 1. Rashid Khan, Anik Das, Build Better Chatbots: A Complete Guide to Getting Started with Chatbots, Apress **REFERENCE BOOKS:** 1. Drexen Braxley, Chat GPT #1 Bible - 10 Books in 1: A Comprehensive Guide to AI: Elevate Your Daily Life, Increase Work Output, Secure Financial Gains, Foster Career Growth, and Cultivate Modern Talents Paperback 2. D. Nardo Publications, ChatGPT Made Simple How Anyone Can Harness AI To Streamline Their Work, Study & Everyday Tasks To Boost Productivity & Maintain Competitive Edge By Mastering Prompt Engineering 3. Robert E. Miller,Prompt Engineering Bible Join and Master the AI RevolutionsProfit Online with GPT-4 & Plugins for Effortless Money Making! 4. Lucas Foster, Chat GPT Bible Developer and Coder Special Edition: Enhancing Coding Productivity with AI-Assisted Conversations.

INTERNAL ASSESSMENT (IA)

Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.

Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

CO's-PO's & PSO's MAPPING

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	3	2.4	2.4	2	2	3	2.6	2

1 - low, 2 - medium, 3 - high, '-' - no correlation

BTDS 832OE: EVOLUTIONARY COMPUTING (Open Elective – III)
B.Tech. IV Year II Sem.

Teaching Scheme:	Credit	Examination Scheme:
TH: - 4 Hours/Week	TH: 04	In Sem. Evaluation: 25 Marks Mid Sem. Exam: 25 Marks End Sem. Exam: 50 Marks Total : 100 Marks

Course Prerequisites: Knowledge on algorithms

Course Objective:
Introduce the concepts of evolutionary computing and various evolution algorithms

Learning Course Outcome:
After successful completion of the course, students will able to:

- Appraise the significance of evolutionary computing
- Apply genetic operators and genetic programming for classification problems
- Hybridization of genetic algorithms with other techniques
- Understand multi objective, interactive evolutionary algorithms

Course Contents

UNIT-I	Optimization, Modelling, and Simulation Problems	07 Hours

Search Problems, Optimization Versus Constraint Satisfaction, The Famous NP Problems
Evolutionary Computing: The Origins: The Main Evolutionary Computing Metaphor, Brief History, The Inspiration from Biology, Evolutionary Computing
Evolutionary Algorithm: Definition, Components of Evolutionary Algorithms, An Evolutionary Cycle by Hand, Example Applications, The Operation of an Evolutionary Algorithm, Natural Versus Artificial Evolution, Evolutionary Computing, Global Optimization, and Other Search Algorithms

UNIT-II	Representation, Mutation, and Recombination	08 Hours

Representation and the Roles of Variation Operators, Binary Representation, Integer Representation, Real-Valued or Floating-Point Representation, Permutation Representation, Tree Representation **Fitness, Selection, and Population Management:** Population Management Models, Parent Selection, Survivor Selection, Selection Pressure, Multimodal Problems, Selection, and the Need for Diversity
Popular Evolutionary Algorithm Variants: Genetic Algorithms, Evolution Strategies,

Evolutionary Programming, Genetic Programming, Learning Classifier Systems, Differential Evolution, Particle Swarm Optimization, Estimation of Distribution Algorithms.

UNIT-III	Parameters and Parameter Tuning	08 Hours

Evolutionary Algorithm Parameters, EAs and EA Instances, Designing Evolutionary Algorithms, The Tuning Problem, Algorithm Quality: Performance and Robustness, Tuning Methods.
Parameter Control: Introduction, Examples of Changing Parameters, Classification of Control Techniques, Examples of Varying EA Parameters.

UNIT-IV	Working with Evolutionary Algorithms	07 Hours

Working of EA, Performance Measures, Test Problems for Experimental Comparisons, Example Applications
Hybridization with Other Techniques: Memetic Algorithms: Motivation for Hybridizing EAs, A Brief Introduction to Local Search, Structure of a Memetic Algorithm, Adaptive Memetic Algorithms, Design Issues for Memetic Algorithms, Example Application: Multistage Memetic Timetabling

UNIT-V	Multi Objective Evolutionary Algorithms	08 Hours

Multiobjective Optimization Problems, Dominance and Pareto Optimality, EA Approaches to Multiobjective Optimization, Example Application: Distributed Coevolution of Job Shop Schedules **Constraint Handling:** Two Main Types of Constraint Handling, Approaches to Handling Constraints, Example Application: Graph Three-Colouring.
Interactive Evolutionary Algorithms: Characteristics of Interactive Evolution, Algorithmic Approaches to the Challenges of IEAs, Interactive Evolution as Design vs. Optimization, Example Application: Automatic Elicitation of User Preferences.

TEXT BOOK:

1. A. E. Eiben, J. E. Smith, Introduction to Evolutionary Computing, Second Edition, Springer.

REFERENCE BOOKS:

1. David E. Goldberg, "Genetic Algorithms in search, Optimization & Machine Learning".
2. Neural Networks and Fuzzy Logic System by Bart Kosko, PHI Publications.

CO’s-PO’s & PSO’s MAPPING

INTERNAL ASSESSMENT (IA)
Two internal assessments and an end semester examination to test students' reading and writing skills along with their grammatical and lexical competence.
Assessment will include following things:

- Internal Assessment: 25 Marks - 1. Attendance
  2. Assignments
  3. Case Study/Mini Project/Presentation
- Mid Semester Exam: 25 Marks
- End Semester Exam: 50 Marks

Cos	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3
1	3	3	3	2	2	3	3	2
2	-	3	3	1	1	3	3	3
3	-	2	2	2	2	3	2	1
4	-	2	2	2	2	3	2	2
5	-	2	2	3	3	3	3	2
Avg	**3**	**2.4**	**2.4**	**2**	**2**	**3**	**2.6**	**2**

1 - low, 2 - medium, 3 - high, '-' - no correlation