EOPC2208 PYTHON LAB. (0-0-3)

Course Objectives:

- To introduce Python programming concepts building on prior knowledge of C and object-oriented programming.
- To enable students to implement algorithms and solve complex problems using Python.
- To utilize Python libraries for data structures, file handling, and visualization.
- To provide exposure to Python's role in modern fields like data analysis, machine learning, and automation.

Course Outcomes:

Upon successful completion, students will be able to:

- Write efficient Python programs leveraging advanced features.
- Apply Python's built-in data structures for problem-solving.
- Use Python libraries to develop solutions in data analysis, visualization, and computational fields.
- Design mini-projects showcasing Python's versatility in real-world applications.

1. **Python Basics**
   - Python environment setup and basic syntax.
   - Variables, data types, and operators.
   - Control statements (if-else, loops) and functions.
   - Comparison of Python syntax with C and OOP concepts.
2. **Python Data Structures**
   - Lists, tuples, sets, and dictionaries.
   - Nested structures and comprehensions.
   - Applications in implementing data structures (stacks, queues, linked lists, etc.) using Python.
3. **File Handling and Exception Management**
   - Reading from and writing to files.
   - Handling different file formats: CSV, JSON.
   - Exception handling and debugging techniques.
4. **Advanced Python Concepts**
   - Object-oriented programming in Python: classes, objects, inheritance, and polymorphism.
   - Lambda functions, map, filter, and reduce.
   - Iterators and generators.
5. **Data Analysis and Visualization**
   - Introduction to NumPy for numerical computations.
   - Using Pandas for data manipulation.
   - Data visualization using Matplotlib and Seaborn.
6. **Algorithmic Problem Solving**
   - Implementation of algorithms (sorting, searching, etc.) using Python.
   - Graph algorithms and dynamic programming concepts.
   - Performance comparison with C implementations.
7. **Mini-Project** Students will apply Python to solve a real-world problem. Suggested topics:
   - Analyzing and visualizing data (e.g., weather patterns, stock market trends).
   - Automating tasks using Python scripts.
   - Simulation of algorithms (e.g., shortest path, scheduling problems).
   - Developing small applications (e.g., file organizer, text processing tools).