6TH SEMESTER BCA			
	SUB CODE		SUB NAME
MAJOR	CORE-I	PAPER-14	Algorithm Design Techniques
	CORE-I	PAPER-15	Project Work
MINOR	CORE-III	PAPER-3	
SEC	PAPER-3		
VAC	PAPER-4		

Core XIV **Semester VI**

Algorithm Design Techniques

Course Objectives:

- To understand the importance of algorithm design.
- To learn ways to analyze algorithms
- To learn about adoption of different algorithmic styles for solving problems

Learning Outcomes:

Upon completion of this course, students will be able to:

- Learn approaches to algorithm analysis & design
- Learn different searching and sorting techniques
- Learn greedy techniques for problem-solving
- Learn graph-based techniques for practical problem-solving

Unit I:

Algorithm specification: Pseudo code, Asymptomatic Analysis, Space complexity and time complexity, Analysis and design of Insertion sort algorithm, Divide and Conquer paradigm, Recurrence relations, Solving Recurrences: Substitution methods, Recursion tree method, and Master method.

Unit II:

Searching and Sorting: Analysis of Linear Search, Binary Search, Merge Sort and Quick Sort, Heap Sort. Hashing: Hash functions, Hash table, Collision resolution: Chaining and Open Addressing (Linear probing, Quadratic probing, Double hashing).

Unit III:

Greedy Technique: General Method, Applications: Fractional Knapsack Problem, Job Sequencing with Deadlines, Huffman Codes.
Dynamic Programming: General Method, Applications: Matrix Chain Multiplication, longest common subsequence, 0/1 Knapsack.

Unit IV:

Graph Algorithms, Topological sort, Minimum Spanning Trees: Prim's and Kruskal's algorithm, Single-source shortest paths: Bellman-Ford algorithm, Dijkstra's algorithm.

Text Book:

- ✓ *Introduction to Algorithms, by Thomas H, Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, PHI.*

Reference Book:

- *Algorithm Design, by Jon Kleinberg, Eva Tardos*.

BCA 6.2 Lab: Algorithm Design Techniques

1. Write C / C++ Programs to implement Insertion Sort
2. Write C / C++ Programs to implement Merge Sort
3. Write C / C++ Programs to implement Quick Sort
4. Write C / C++ Programs to implement Heap Sort
5. Write C / C++ Programs to implement Hashing
6. Write C / C++ Programs to implement Fractional Knapsack
7. Write C / C++ Programs to implement Matrix Chain Multiplication
8. Write C / C++ Programs to implement Longest Common Subsequence
9. Write C / C++ Programs to implement Huffman Code
10. Write C / C++ Programs to implement Prim's Algorithm
11. Write C / C++ Programs to implement Krushkal's Algorithm
12. Write C / C++ Programs to implement Dijkstra's Algorithm

Core XV

Project Work-I

A student has to do a Project work under the guidance of a faculty member. After completing the project, the student has to submit a project report which has to be evaluated by an external examiner. The model template for the project report can be as follows

1. Title of the project
2. Declaration (by the student)
3. Certificate (of the project guide)
4. Acknowledgement
5. Abstract
   [Provide a brief summary of your project, including its objectives, methods, and key findings.]
6. Table of Contents
   Introduction
   Literature Review
   Methodology
   Results
   Discussion
   Conclusion
   References
7. Introduction
   [Describe the background and context of your project, including the problem statement and objectives.]
8. Literature Review
   [Review relevant literature related to your project, discussing previous research, theories, and concepts.]
9. Methodology/
   [Explain the methods you used to conduct your research or project, including data collection, analysis techniques, and any tools or software used.]
10. Implementation/Software development
11. Results
    [Present the findings of your research or project, using tables, figures, or graphs as needed to illustrate key points.]
12. Discussion
    [Interpret your results and discuss their implications, relating them back to your research objectives and the broader context of your field.]
13. Conclusion
    [Summarize the main findings of your project and their significance, as well as any recommendations for future research or applications.]
14. References
    [List all sources cited in your project using a consistent citation style (e.g., APA, MLA).]

The evaluation pattern of the project will be as follows:

i. Problem formulation and definition
ii. Execution of code & results
iii. Documentation
iv. Clarity in presentation
v. Performance in the Viva voce