

750721, Algorithms and Complexity

3 hours per week, 3 credit hours, prerequisite: **none**

Teaching Method: 37 hours Lectures (2-3 hours per week) + 8 hours Seminars (1 per 2 weeks)

Aims: This module aims to introduce the concepts of analysis and design of algorithms and measure their complexity through different strategies.

Learning Outcomes:

On completion of this module, the student should:

- Understand different strategies of algorithms design.
- Be able to analyze different algorithms.
- Be able to measure the complexity of algorithms.
- Be able to choose the best algorithm for solving a problem.

Textbooks and Supporting Materials:

1. T.H. Cormen, C.E. Leiserson, R .L. Rivest, Introduction to Algorithms, PHI,
2. D. Kreher and D.Stinson: Combinatorial Algorithms: generation, enumeration and search, CRC Press, 1998.
3. Jean-Daniel Boissonnat, Mariette Yvinec, Algorithms Geometry,
4. M. R. Garey, D. S. Johnson, Computers and Intractability: a guide to the theory of NPcompleteness, W. H. Freeman, 1979
5. Frank Thomson Leighton, Introduction to Parallel algorithms and Architectures: Arrays, Trees, Hypercubes,
6. Mark de Berg, Marc van Kreveld, Mark Overmars, Computational Geometry,
7. A. V. Aho, J. E. Hopcroft, J.D. Ulman, The Design and Analysis of Computer Algorithms, Addison Wesley,
8. Michael Seul, Lawrence O’Gorman, Michael J. Sammon, Practical Algorithms for Image Analysis:

Descriptions, Examples, and Code, Cambridge University Press,

Synopsis:

1. Revision of Design Methods: Divide and Conquer design, Dynamic Programming, Greedy algorithm etc. Master method, Order statistics, Dynamic Order Statistics, Red-Black trees, Binomial Heaps and Fibonacci Heaps, Data Structures for Disjoint Sets.

2. Matrix Chain Multiplication, Matrix Operations algorithms, Longest Common Subsequence

Problems

10

3. Graph Algorithms, Minimum spanning Trees, Single-Source Shortest Paths, All-Pairs Shortest Paths, Maximum Flow Algorithms.

4. Sorting Networks, and complexity

5. Polynomials and DFT, FFT algorithms.

6. Combinatorial Algorithms, Number Theoretic Algorithms, String Matching algorithms,

7. Cryptography

8. Computational Geometry, Algorithmic Geometry and Complexity.

9. NP-Completeness, Non-deterministic algorithms

10. Exponential-time Techniques

11. Approximation algorithms

12. Parallel Algorithms.

Assessment: Two 1-hour midterm exams (15% each); Assignments (10%); Seminars (10%); 2-hours

Final Exam (50%)