**ADVANCED ALGORITHMS**

**A.A. 2017-2018**

***Prof.ssa Rossella Petreschi***

**Advanced Analysis Techniques**

**(Ref.3/Chap.17)**

**AMORTIZED ANALYSIS:** The average performance of each operation in the worst case. The Aggregate Analysis: assigning the same cost to different operations. The Accounting method: how to assign a credit to a specific object. The potential function and the amortized cost of an operation with respect to a potential function. Illustration of the amortized analysis: operations on stack and the increment operation on a binary counter. The management of a dynamic table: tables expansion and tables contraction.

**Data Structure for Set Manipulation Problems**

**(Ref.3/Chap.12,14,18,19,21; Ref.7/Chap.7; Ref.8/Chap.6)**

**SYMBOL TABLES:** Binary search tree. Visit a binary search-tree. Insertion and deletion in a binary search-tree. Average analysis of sequence of insertion operations in a binary search-tree. Balanced search-trees. AVL-trees. The height of an AVL-tree is logarithmic. Rotations on AVL-trees. Insertion and deletion in AVL- trees. Self-adjusting trees. Splay-operation. Make a binary search tree a splay-tree. Amortized analysis of a single splay-step. Amortized analysis of a sequence of operations on a splay-tree. Definition of B-trees. B-trees height. Basic operations on B-trees. Deleting a key from a B-tree.

**AUGMENTING DATA STRUCTURES:** How to augment a data structure. Structure of Fibonacci heaps (FH). Inserting an element in a FH. Decreasing a key in a FH. Deleting a node in a FH. Merge two FH. Computing the amortized analysis of all the operations on a FH. Comparing heaps and Fibonacci heaps. Bounding the maximum degree in a FT. Dijkstra's algorithm for single source shortest paths. Management of the rank of an element in an AVL tree. Interval trees. Search, insert and delete in an interval tree. Correctness of the interval search procedure.

**DATA STRUCTURES FOR DISJOINT SETS:** Union by size and Union by rank. Balanced Quick-union trees and Balanced Quick-find trees. Forests of disjoint sets. Heuristics to improve running times. Union by rank through path compression.

**Efficient Algorithms for Domain-Specific Problems**

**(Ref. 1/Chap.16,17; Ref. 9/Chap. 1,3)**

**MATCHING:** Maximal, maximum and perfect matching. Alternating and augmenting paths. XOR operator and its properties. The Hungarian tree method for bipartite graphs. Blossom's contraction and expansion in general graphs. Algorithms to find a maximum matching in a general graph and in a bipartite graph.

**NETWORK FLOW**: Transportation networks. Flow on a network. Pushing flow on forward and backward edges. Residual networks method.Augmenting path and bottleneck.Ford and Fulkerson' algorithm. Max-flow and min-cut. Networks with multiple sources and multiple tails. Bipartite graphs: matching and flow network.

**PLANARITY:** Planar graphs.Euler's formula.Kuratowski’s theorem.st-numbering.Functions DFN, FATHER and LOW.Function PATH.Algorithm for st-numbering. Bush forms. PQ-trees.

**Parallel and Distributed Algorithms**

**(Ref. 2/Chap. 3; Ref. 4; Ref. 5/Chap.1,2,4,5; Ref. 6/Chap. 12)**

**ISSUES IN PARALLEL AND DISTRIBUTED COMPUTING:**

Concurrent system vs sequential system. Distributed system vs parallel system. Synchronous and asynchronous systems. Shared memory and concurrent write and read. EREW vs CREW. Speed up and efficiency. Brent's theorem on CREW and on EREW. Decreasing the number of processors. Interconnection networks: Mesh. Binary Tree and Hypercube

**INTERCONNECTION NETWORKS:**

Zero-One Principle. Bi-tonic Sequences. Bi-tonic Merge Circuit. Bi-tonic Sorting Network.

Sorting Networks. Mesh Model. Binary Tree Model

**BASIC PARALLEL ALGORITHMS:** First half technique: compute sum and find maximum. Pointer Jumping Technique. List ranking. Prefix Sums. EREW vs CREW: broadcast on P-RAM, Mesh and Binary Tree. Sum and prefix sum on P-RAM, Mesh and Binary Tree. Accelerated Cascading technique. Compute sum with accelerated cascading. The Euler tour technique: rooting a tree, finding the root; postorder and preorder numbering; computing the vertex level.

An advanced example of parallel algorithm: Minimum Spanning Tree Searching.

**BASIC DISTRIBUTED ALGORITHMS:** General scheme of a distributed algorithm. Broadcast on a distributed ring. Broadcast with echo in a general graph. Leader election on a ring: n initializers. Leader election on a ring: one initializer. Leader election on a ring: k-neighbourly.

An advanced example of distributed algorithm: Minimum Spanning Tree Searching.

**References:**

Alsuwaiyel M.H.,

***"Algorithms. Design Techniques and Analysis"***, World Scientific

Attiya H.,Welch J.

***"Distributed Computing"***, McGraw-Hill

Cormen T.H., Leiserson C.E., Rivest R.L, Stein C.,

**"*Introduction to algorithms"***, MIT Press

Gallager R.G., Humblet P.A., Spira P.M.,

***"A Distributed Algorithm for Minimum-Weight Spanning Trees"*** [gallager.....pdf](http://twiki.di.uniroma1.it/pub/AA/AdvancedAlgorithms_2016_17/gallager.....pdf)

Jaja J.,

***"An Introduction to Parallel Algorithms"***, Addison-Wesley Pub.Comp.

Johnsonbaugh R.,Schaefer M.,

***"Algorithms"*,** Pearson Educational International

Kingston J.K.,

***"Algorithms and data Structures: Design, Correctness, Analysis"***, Addison-Wesley Pub.Comp.

Levitin A.,

***"The design and analysis of algorithms"***, Addison-Wesley Pub.Comp.

Nishizeki T., Chiba N.,

***"Planar graphs:theory and algorithms"*** , North-Holland