CSCE-629 Analysis of Algorithms

Fall 2017

Instructor: Dr. Jianer Chen
Office: HRBB 315C
Phone: 845-4259
Email: chen@cse.tamu.edu
Office Hours: T,Th 2:00pm-3:30pm

Teaching Assistant: Qin Huang
Office: HRBB 315A
Phone: 402-6216
Email: huangqin@email.tamu.edu
Office Hours: T,Th 9:00am-12:00noon

Assignment # 3
(Due October 26, 2017)

1. Design efficient algorithms for the following problems:
(1) given a undirected graph G, test if G is a tree;
(2) given a unweighted and undirected graph G, and a vertex s in G, compute the distance from s to every other vertex in G.

2. In the linear-time algorithm $Rank(S, k)$ we discussed in class, which returns the kth smallest element of the set S, we first divided the input set S into groups of 5 elements. Show that if we modify the algorithm by using groups of 7 elements instead of groups of 5 elements, the algorithm still works and runs in linear time. Discuss which version you would use in practice.

3. Modify the QuickSort algorithm so that the pivot is selected using the linear-time Median-Finding algorithm. Prove that this modified QuickSort algorithm takes time $O(n \log n)$ *in the worst case*. Discuss why this algorithm is not used in practice.

4. This question lets you to work out the details for an algorithm we discussed in class. Suppose that we have a weigthed graph G, where multiple edges between vertices are allowed. Assume that the graph G is given in the adjacency list structure $A[1..n]$. Therefore, for each vertex v, the linked list $A[v]$ for v may contain elements of the forms $[u, w_1]$ and $[u, w_2]$, standing for two edges from v to u, with weights w_1 and w_2, respectively. Now the goal is that for each pair of vertices in the graph G, we want to remove all multiple edges except the one with the largest weight. Design a linear-time algorithm for this problem.