TEL AVIV UNIVERSITY
Department of Computer Science
0368.4281 – Advanced topics in algorithms
Spring Semester, 2013/2014

Homework 3, May 2, 2014

Due on May 18. Please submit a pdf electronically.

1. Consider the problem of maintaining a minimum spanning forest when we insert and delete edges (fully dynamic MSF problem) between pairs of a fixed set of n vertices. Find a data structure that allows to insert and delete edges in $O(n \log n)$ worst case time per insertion and deletion. Your data structure should take $O(n^2)$ space and be as simple as you can.

2. Show how to improve the order maintenance data structure presented in class such that the amortized time per operation is O(1).

3. Consider a variation of the incremental cycle detection problem in which we explicitly maintain the vertices in an **array** A in topological order as long as the graph is acyclic, and report a cycle when one is formed. Describe an algorithm that takes $O(mn)$ time for m insertions.

4. We say that a graph G is k-separable if the vertices of **every subgraph** of G of m edges can be partitioned into three sets A, S, B, such that 1) S is of size at most k. 2) There is no edge between a vertex in A and a vertex in B. 3) The number of edges incident to vertices of A is at most $2m/3$ and the number of edges incident to vertices of B is at most $2m/3$.

Consider the incremental cycle detection problem where we are guaranteed that at any time the graph is k-separable.

Prove that the bidirectional cycle detection algorithm presented in class processes such sequences in $O(km \log^2 n)$ time. (Hint: Classify the insertions into 2 types: those that change the position – in the topological order – of a vertex in S and those that do not. Use induction to bound insertions that do not move vertices in S.)