Algorithms: MidSem Exam

Duration: 3 hours

September 29, 2017

Answer all questions (Maximum marks 60).

1. Design a linear time algorithm to detect the presence of an edge such that removing the edge from the graph leaves behind a connected graph. Prove its correctness. *(5 marks)*

2. Let T be a rooted k-ary tree i.e. each internal node has at most k children. Give a linear time algorithm to find an edge e of the graph such that deleting e leads to two subtrees neither of which contains more than k times the vertices in the other subtree. *(10 marks)*

3. Let G be a directed graph with edges labelled by rational numbers called weights. The weight of a directed cycle is the product of weights of the directed edges of the cycle. Describe an efficient algorithm to check if there is a directed cycle in the graph with weight greater than 1. How will you find such a cycle? What is the number of arithmetic operations that your algorithm needs? Notice that you are only allowed to add/subtract/multiply/divide/compare rationals in unit time. *(15 marks)*

4. You are given n objects $a_1, \ldots, a_n$. You also have access to a function f that takes in two objects and declares if they are equivalent. The function f defines a relation that is an equivalence relation i.e. it is reflexive, symmetric and transitive. Show that by $O(n \log n)$ invocations of f you can determine if a majority (i.e. more than $n/2$) of the objects are equivalent. *(15 marks)*

5. Given an array A of length n, give an efficient algorithm to find the longest increasing subsequence in A and analyse its complexity. An increasing subsequence of length k is a set of indices $0 = i_1 < i_2 < \ldots < i_k < n$ such that, $A[i_1] < A[i_2] < \ldots < A[i_k]$. *(15 marks)*