KTH Computer Science
and Communication

Homework II, Approximation Algorithms 2010

Due on Tuesday October 26 at 13.15 (hand in at start of lecture or send an email to osven@kth.se). Solutions to many homework problems, including problems on this set, are available on the Internet, either in exactly the same formulation or with some minor perturbation. It is *not acceptable* to copy such solutions. It is hard to make strict rules on what information from the Internet you may use and hence whenever in doubt contact Ola Svensson. You are, however, allowed to discuss problems in groups with up to three students, but solutions should be handed in individually.

1 (30 p) Rumors say that the renowned scientist Skalman has come up with a r-approximation algorithm for Weighted Vertex Cover on *triangle-free* graphs, for some $r < 2$. (Triangle-free graphs are those graphs that do not contain three vertices v_1, v_2, v_3 that are all adjacent.)

Show that Skalman's result in fact implies a $\max[r, 1.5]$-approximation algorithm for Weighted Vertex Cover on general graphs.

2 (30 p) A Super-PTAS is defined like an FPTAS but the running time is polynomial in the size of the instance and $\log(1/\epsilon)$, that is, the number of bits needed to represent ϵ. Prove that if a problem with integer-valued objective function that can be evaluated in polynomial time has a Super-PTAS, then it can be solved in polynomial time.

Hint: This is similar to the proof that (most) strongly NP-hard problems do not admit an FPTAS (see Section 8.3 in Vazirani's book).

3 (40 p) Consider the Minimum Makespan Scheduling Problem on a constant number m of machines.

3a (20 p) Give a pseudo-polynomial algorithm based on dynamic programming that runs in time $O(n \cdot p_{sum}^m)$ where p_{sum} is the sum of all processing times.

3b (20 p) Use the pseudo-polynomial algorithm to obtain an FPTAS.

Hint: Start by considering 2 machines and then generalize to more machines.