Anna Adamaszek, Andreas Wiese
Tutorials: Marvin Künnemann **Summer 2014**

Assignment 1 for
Approximation Algorithms and Hardness of Approximation

Discussion:
Thursday, 24 April 2014, 14 pm

Assignment 1 *(Vertex Cover on Trees)*

Show that on trees, one can find an optimal vertex cover in polynomial time.

Assignment 2 *(Vertex Cover: Tightness)*

In the lecture, we have discussed two approximation algorithms for Vertex Cover and proved that they have an approximation performance of at most 2. For both algorithms, find instances on which they give solutions that are not better than a 2-approximation, i.e., the algorithm gives no α-approximation for any $\alpha < 2$.

Assignment 3 *(Bin Packing)*

Give examples showing that

1. NextFit is not an α-approximation for any constant $\alpha < 2$.
2. FirstFitDecreasing is not an α-approximation for any constant $\alpha < \frac{3}{2}$.

Assignment 4 *(Greedy Knapsack)*

The following strategy is a simple greedy approach to approximate the Knapsack problem: Sort all items in decreasing order of their ratios of profit and size, i.e., $\frac{p_1}{s_1} \geq \frac{p_2}{s_2} \geq \cdots \geq \frac{p_n}{s_n}$. In this order, we test for each item i if it still fits into the knapsack. If yes, we include i into our solution. Then we proceed with the next item and stop when all items have been tested.

1. Show that this algorithm can perform arbitrarily bad, i.e., for any $B > 1$, we can find an instance on which it gives nothing better than a B-approximation.
2. Suppose we modify this algorithm slightly: After computing the greedy solution, we compare this solution to taking only the most profitable item that fits into the knapsack. If taking only this item yields a higher profit, we forget our greedy solution and use this solution instead. Show that this simple modification gives a 2-approximation to the Knapsack problem.