CS261: Exercise Set #8

For the week of February 22–26, 2016

Instructions:

(1) *Do not turn anything in.*

(2) The course staff is happy to discuss the solutions of these exercises with you in office hours or on Piazza.

(3) While these exercises are certainly not trivial, you should be able to complete them on your own (perhaps after consulting with the course staff or a friend for hints).

Exercise 36

Recall the MST heuristic for the Steiner tree problem — in Lecture #15, we showed that this is a 2-approximation algorithm. Show that, for every constant $c < 2$, there is an instance of the Steiner tree problem such that the MST heuristic returns a tree with cost more than c times that of an optimal Steiner tree.

Exercise 37

Recall the greedy algorithm for set coverage (Lecture #15). Prove that for every $k \geq 1$, there is an example where the value of the greedy solution is at most $1 - (1 - \frac{1}{k})^k$ times that of an optimal solution.

Exercise 38

Recall the MST heuristic for the metric TSP problem — in Lecture #16, we showed that this is a 2-approximation algorithm. Show that, for every constant $c < 2$, there is an instance of the metric TSP problem such that the MST heuristic returns a tour with cost more than c times the minimum possible.

Exercise 39

Recall Christofides's $\frac{3}{2}$-approximation algorithm for the metric TSP problem. Prove that the analysis given in Lecture #16 is tight: for every constant $c < \frac{3}{2}$, there is an instance of the metric TSP problem such that Christofides's algorithm returns a tour with cost more than c times the minimum possible.

Exercise 40

Consider the following variant of the traveling salesman problem (TSP). The input is an undirected complete graph with edge costs. These edge costs need *not* satisfy the triangle inequality. The desired output is the minimum-cost cycle, not necessarily simple, that visits every vertex *at least* once.

Show how to convert a polynomial-time α-approximation algorithm for the metric TSP problem into a polynomial-time α-approximation algorithm for this (non-metric) TSP problem with repeated visits allowed.

[Hint: Compare to Exercise 32.]