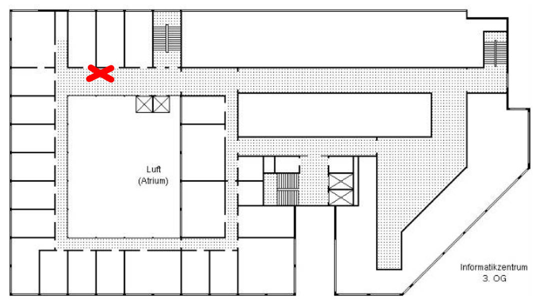
Abteilung Algorithmik
Institut für Betriebssysteme
und Rechnerverbund
TU Braunschweig

SoSe 15

Prof. Dr. Sándor Fekete
Dr. Victor Alvarez
Melanie Papenberg

Approximation Algorithms
Homework Set 2, 11. 05. 2015

Solutions are due on Monday, June 1st, 2015, until 13:00 in the cupboard for handing in practice sheets. **Please put your name on all pages!**



Exercise 1 (Greedy Set Cover algorithm): Apply the Greedy Set Cover Algorithm (Algorithm 2.3 from the lecture) to the following Set Cover instance:

Cost function $c : S \to \mathbb{Q}^+$ defined as $c(S_i) = |S_i| + 1$,
Universe: $U = \{1, \ldots, 20\}$, and
$S = \{S_1, \ldots, S_{11}\}$ where each S_i is defined as:

$S_1 = \{1, 2, 3, 4\}$
$S_2 = \{5, 6, 7, 8\}$
$S_3 = \{9, 10, 11, 12\}$
$S_4 = \{13, 14, 15, 16\}$
$S_5 = \{17, 18, 19, 20\}$
$S_6 = \{1, 2, 3, 5, 6, 7, 9, 10, 11\}$
$S_7 = \{14, 15, 16, 17, 18, 19\}$
$S_8 = \{12, 13, 14, 15\}$
$S_9 = \{4, 5, 6\}$
$S_{10} = \{7, 8, 9\}$
$S_{11} = \{18, 19, 20\}$.

If the maximum in step 2 of the algorithm is not uniquely defined, choose the set S_i with minimum index.

What is the value of the computed set cover? Can you give a better set cover? **(10 pts.)**

Exercise 2 (Complexity classes): Explain in your own words what it means that a (computational) problem P is:

(1) in NP.
(2) NP-complete.
(3) NP-hard.

(4) Is the problem of *computing* the smallest spanning cycle of a graph (TSP) NP-complete or NP-hard? **(2.5+2.5+2.5+2.5 = 10 pts.)**

Exercise 3 (Christofides' algorithm): Consider the following approximation algorithm:

Algorithm:
Input: A complete weighted graph $G = (V, E)$ with a metric distance function $d : V \times V \to \mathbb{R}^+$.
Output: A spanning cycle C of length L, such that $L/L^* \leq 3/2$, where L^* is the length of the shortest spanning cycle of G.

1. Compute a minimum-weight spanning tree T of G.
2. Construct the set V' of vertices of odd degree in T and find a minimum-weight *perfect* matching M of V' in G.
3. Construct the Eulerian graph G' obtained by adding the edges of M to T.
4. Find an Eulerian cycle C' of G' and index each vertex according to the order, $ind(v)$, in which v is first visited while traversing C'.
5. Output the following approximate minimum-weight spanning cycle: $C = (v_1, v_2, v_3, \ldots, v_n, v_1)$ where $ind(v_i) = i$.

Give a formal proof that Christofides' algorithm provides a $3/2$-approximation. **(20pts.)**

Exercise 4 (Tightness of approximation algorithms for TSP)

1. Show that the approximation factor of the twice-around-MST-algorithm is tight. That is, show how to construct arbitrarily large instances for which the algorithm produces an approximation that is twice as large as the optimal solution. You may assume that the resulting approximation factor is $2 - \varepsilon$, where $\varepsilon \to 0$ as $n \to \infty$.
   Hint: Consider a graph with edge-weights 1 and 2 in which *one* of the minimum-weight spanning trees is a star.
2. Show that the approximation factor of Christofides' algorithm is tight. As before, show how to construct arbitrarily large instances for which the algorithm produces an approximation factor of $3/2$. You may assume that the resulting approximation factor is $3/2 - \varepsilon$, where again $\varepsilon \to 0$ as $n \to \infty$.
   Hint: Consider a properly weighted graph in which *one* of the minimum-weight spanning trees is a path.

In the instances you show, the triangle inequality must hold. **(10 + 10 pts.)**