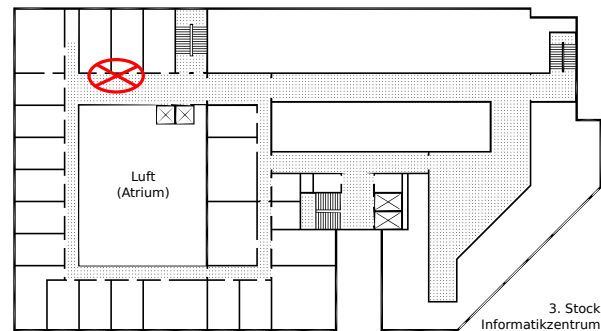


Prof. Dr. Sándor P. Fekete
Phillip Keldenich

Online Algorithms

3rd Homework Assignment, 28th of May 2018

Solutions are due Monday, the 11th of June 2018, until 1:15 PM in the homework cupboard. You can also hand in your solution in person before the small tutorial begins. If you cannot hand in the homework in person, you can also hand it in via e-mail to both j.heroldt@tu-bs.de and keldenich@ibr.cs.tu-bs.de. Please clearly label your solutions using your name and matriculation number.



Exercise 1 (Online Bin Covering): In this exercise, we consider the problem of BIN COVERING in an online scenario. Analogous to the situation for online bin packing, we are given a sequence of items of unknown weights $a_1, \dots, a_n \in [0, 1]$ and want to assign these items to bins in an online fashion; however, the bins do not have limited capacity. In the BIN COVERING problem, we want to *maximize* the number of *covered* bins, i.e., the number of bins that receive items of total weight at least 1.

- Find an online algorithm for BIN COVERING with an absolute competitive ratio of 2 and prove its competitive ratio. Prove that no deterministic online algorithm can have an absolute competitive ratio $c < 2$.
- Prove that no deterministic online algorithm for BIN COVERING can have an asymptotic competitive ratio $c < 3/2$.

(10+10 points)

Exercise 2 (Online Coloring): In this exercise, we consider the problem of GRAPH COLORING in an online scenario. Our input sequence consists of the vertices v_1, \dots, v_n of an undirected graph $G = (\{v_1, \dots, v_n\}, E)$. Together with each vertex v_i , we are given the list E_i of all edges connecting v_i to previously given vertices v_1, \dots, v_{i-1} . Edges from v_i to vertices v_j with $j > i$ are only revealed once vertex v_j is given. The number of edges and vertices in the graph is not known in advance.

When we are given v_i , we have to choose a color $c(v_i) \in \mathbb{N}$ for v_i in such a way that no vertex adjacent to v_i has color $c(v_i)$. As usual, this choice is final; we cannot change

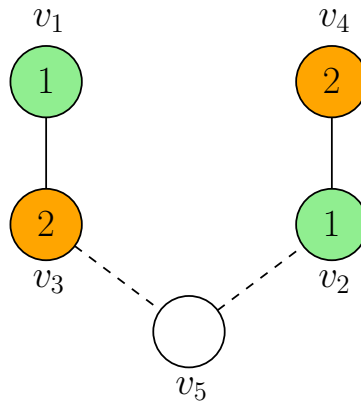


Figure 1: A snapshot from the execution of an online graph coloring algorithm. The algorithm has colored the vertices v_1, \dots, v_4 using colors 1 and 2 and is now given a new vertex v_5 connected to v_2 and v_3 . It cannot use color 1 or 2 for v_5 and therefore has to introduce a new color for v_5 .

the color later on. We want to minimize the number of colors used. For an example, see Figure 1.

We want to show that there is no deterministic c -competitive algorithm for this problem for any constant c . For any constant number $2 \leq k \in \mathbb{N}$ of colors and any deterministic online algorithm \mathcal{A} , devise a strategy for an adversary that satisfies the following requirements.

- The strategy always produces a forest $T_{\mathcal{A},k}$.
- The online algorithm \mathcal{A} uses at least k colors on $T_{\mathcal{A},k}$.

Offline, every forest can be colored with 2 colors; therefore, this implies that \mathcal{A} 's competitive ratio is at least $k/2$. **(20 points)**

Exercise 3 (Line search): Suppose that you are a hungry cow on a path. You know that at some distance $D \geq 1$ along the path there is a pasture with tasty grass; however, you do not know the distance D or the direction in which you have to go. In order to reach the pasture, you start by traveling into one of the two directions for one unit of distance. If you do not find your food, you return to the start and then travel two units of distance into the other direction. If you still have not found your food, you return to the start and then travel four units of distance into the first direction, and so on. You repeat this procedure, doubling the distance each time you return back to the start, until you finally reach your goal.

Prove that this strategy is 9-competitive, i.e., that you never have to travel more than $9D$ units of distance until you reach the goal. Moreover prove that this is tight, i.e., that there is no constant $c < 9$ such that this strategy is c -competitive. **(20 points)**