Advanced and algorithmic graph theory

Summer term 2019

Exercise sheet 5

Exercises for the exercise session on 27/05/2019

Problem 5.1. Prove that the recursive-largest-first algorithm colours all bipartite graphs optimally and show that it can be implemented to run in time $O(nm)$.

Problem 5.2. Given a non-empty graph G, denote by $P_G\colon \mathbb{N} \to \mathbb{N}$ the function that maps each $k \in \mathbb{N}$ to the number of k-colourings of G (recall that we assume the set of colours of a k-colouring to be $\{1, \dots, k\}$).

(a) Use induction on $\|G\|$ to prove that P_G is a polynomial of the form

$$P_G(k) = k^{|G|} - \|G\|\, k^{|G|-1} + \sum_{i=1}^{|G|-2} a_i k^i.$$

(P_G is also called the *chromatic polynomial* of G.)

(b) Describe how to determine the chromatic polynomial of a graph algorithmically. What running time do you need?

Problem 5.3. Prove directly (that is, without using any results about edge-colourings from the lecture) that every k-regular bipartite graph is k-edge-colourable. Prove that this implies Theorem 4.22, i.e. $\chi'(G) = \Delta(G)$ for every bipartite graph.

Problem 5.4. Describe an algorithm that finds, for every input graph G, an edge-colouring of G with at most $\chi'(G) + 1$ colours. What running time can you achieve?

Hint. Vizing's theorem and its proof.

Problem 5.5. For every $k \in \mathbb{N}$, construct a bipartite graph G_k and an assignment of lists that shows that G_k is *not* k-choosable.

Problem 5.6. A *total colouring* of G is a function $c\colon V(G) \cup E(G) \to S$ such that $c|_{V(G)}$ and $c|_{E(G)}$ are vertex- and edge-colourings, respectively, and in addition no edge has the same colour as one of its end vertices. We write $\chi_t(G)$ for the least k for which there exists a total colouring of G with k colours.

Prove that the list colouring conjecture would imply $\Delta(G)+1 \leq \chi_t(G) \leq \Delta(G)+3$. (The *total colouring conjecture* asserts that even $\chi_t(G) \leq \Delta(G) + 2$.)