NDMI012: Combinatorics and Graph Theory 2
HW#5

Irena Penev
Winter 2020/2021

due Tuesday, April 13, 2021 before midnight (Prague time)

Remark: Please e-mail me (ipenev@iuuk.mff.cuni.cz) your HW as a **PDF attachment** (no other format will be accepted).

Problem 1 (25 points)**.** *Let n be a positive integer, let $a_1, \ldots, a_n, b_1, \ldots, b_n \in \mathbb{R}$ be such that $a_1 < b_1, \ldots, a_n < b_n$, and let $I_1 = (a_1, b_1), \ldots, I_n = (a_n, b_n)$. Let G be the graph with vertex set $\{v_1, \ldots, v_n\}$, and with adjacency as follows: for all distinct $i, j \in \{1, \ldots, n\}$, v_i is adjacent to v_j in G if and only if $I_i \cap I_j \neq \emptyset$.*[1] *Prove that G is a chordal graph.*

> ***Hint:*** *Induction on n. In the induction step, find a simplicial vertex, delete it, and apply the induction hypothesis.*

Problem 2 (25 points)**.** *Let T be a tree, and let $T_1, \ldots, T_k$, with $k \geq 1$, be (not necessarily distinct) subtrees of T. Assume that for all distinct $i, j \in \{1, \ldots, k\}$, $V(T_i) \cap V(T_j) \neq \emptyset$. Prove that $V(T_1) \cap \cdots \cap V(T_k) \neq \emptyset$.*

> ***Hint:*** *Induction on $|V(T)|$.*

Problem 3 (25 points)**.** *Let T be a tree, and let $T_1, \ldots, T_k$, with $k \geq 1$, be (not necessarily distinct) subtrees of T. Let G be the graph with vertex-set $V(G) = \{v_1, \ldots, v_k\}$, and adjacency as follows: for all distinct $i, j \in \{1, \ldots, k\}$, $v_i v_j \in E(G)$ if and only if $V(T_i) \cap V(T_j) \neq \emptyset$. Prove that G is chordal.*

> ***Hint:*** *Induction on $|V(T)|$.*

[1] A graph obtained in this way is called an *interval graph*.

Problem 4 (25 points)**.** *Let G be a chordal graph, and set $V(G) = \{v_1, \ldots, v_n\}$. Prove that there exists a tree T and (not necessarily distinct) subtrees $T_1, \ldots, T_n$ of T such that for all distinct $i, j \in \{1, \ldots, n\}$, $v_i v_j \in E(G)$ if and only if $V(T_i) \cap V(T_j) \neq \emptyset$.*

> ***Hint:*** *Induction on n. In the induction step, delete a simplicial vertex of G, and then apply the induction hypothesis. Now you need to modify the tree and the subtrees that you got, and create a subtree corresponding to the vertex you deleted. You may use Problem 2.*