# HOMEWORK 6

A *graph* is a set of numbered points (called *vertices*) some of them are joined by lines (called *edges*). It is important which vertices are joined by an edge and which are not; all the rest (mutual position of vertices, form of edges, their intersections, etc.) is irrelevant.

A graph is called connected if any two its vertices can be joined by a path going along the edges. Such a path is called a cycle if its endpoints coincide and it never goes twice along the same edge. A connected graph containing no cycles is called a *tree*. We will suppose that a tree has $n+1$ vertices numbered $0, 1, \dots, n$.

**Problem 1.** List all trees with 3 and 4 vertices.

A vertex is called hanging if it is an endpoint of one edge only.

**Problem 2.** (a) Prove that a tree contains at least one hanging vertex. (b) Prove that if we delete a hanging vertex together with the corresponding edge then a tree will remain a tree. (c) Prove that any tree with $(n+1)$ vertices contains $n$ edges.

**Problem 3.** A coloring of the vertices of a tree is called regular if every two vertices joined by an edge have different color. How many are there ways to color a tree using $k$ different colors ? (there is no condition that all the colors should be used)

**Problem 4.** Prove that any two vertices of a tree can be joined by a unique path which never goes twice along the same edge.

To every tree we can associate a sequence $b_1, \dots, b_{n-1}$ by the following law. Let $A$ be a hanging vertex with the biggest number; it is attached to some vertex $B$; let $b_1$ be the number of $B$. Delete now the vertex $A$ together with the edge $AB$; then repeat the procedure to get $b_2$, etc.

**Problem 5.** Prove that one can restore the tree uniquely from the sequence $b_1, \dots, b_{n-1}$.

**Problem 6.** Using Problem 5 count the number $T_n$ of different trees (with $n+1$ vertices).

A tree is called monotone if for every $k > 0$ the path joining the vertex number $k$ with the vertex number 0 passes only vertices with the numbers smaller than $k$.

**Problem 7.** (a) How to understand from the sequence $b_1, \dots, b_{n-1}$ whether the tree is monotone or not? (b) Count the number of monotone trees (with $n+1$ vertices).

One says that a tree has 1 inversion if the monotonicity condition is violated for exactly one vertex.

**Problem 8.** (a) Using the result of Problem 7 prove that the number of trees with 1 inversion such that $n$ is not a hanging vertex is $(n-1)(n-1)!$. (b) Prove that the total number of trees with 1 inversion (and $(n+1)$ vertices) is $n!(n-1)/2$.