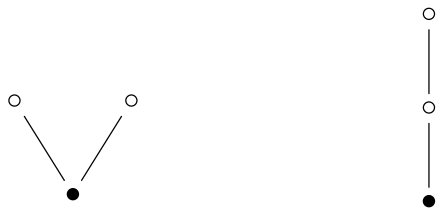
Project: Forests without odd symmetries

Supervisor: Markus Szymik

Background: A tree is a connected graph with no loops. A rooted tree is a tree with a chosen vertex. A forest is a set of rooted trees. Each rooted tree or forest has a group of symmetries. Sometimes this group contains a symmetry that induces an odd permutation of the vertex set, and sometimes it does not.



Problem: What is the number of forests with n vertices that have no odd symmetry?

Specification: (1) Find a way to produce a list of all rooted trees with n vertices. (2) Go through the list to check which have odd symmetries and which have not. (3) Count! (4) Assemble the information for trees to make a statement about forests.

This can (and should) be done for small n by hand, and for larger n with a computer. Given n, let it compute the number asked for, and ideally also pretty-print the trees and forests.

(5) Formulate and prove statements about the resulting sequence of numbers.

Prerequisites: There is not much required to understand the problem: The graph theory is almost completely explained above, and you need to be able to distinguish between even and odd permutations. As for the computational part, some software skills are required. There exists some packages to find the symmetry groups of graphs, and also some algorithms to list all rooted forests with n vertices. In order to combine these, a litte bit of coding might be necessary.

Training: Because the statement of the problem is so elementary, we could start to work towards a solution right away.

Time frame: The time frame for this project is around 100 working hours.