# Project Concepts of Data Science 2024-2025

## Description

The purpose of the project is the implementation of a ternary search tree. The concept of this data structure was explained in the second session of the course. You can find additional information on this data structure online, for instance on Wikipedia.

1. The development of the software should be done using version control hosted on GitHub. If your repository is private, do not forget to give me access. You should also follow up on whether the invitation has been accepted, since such invitations easily get lost. Since you are working in teams of two, GitHub is also your tool to collaborate within the team. You are expected to collaborate while developing the code, so that should be reflected by the commit messages for the repository. If there are no or few commit messages by a team member, I will conclude that this person contributed little to the code and that will be reflected by that person's grade. The repository should also contain a README file that documents the content of the repository, as well as a summary of your conclusions.
2. The ternary search tree should be either implemented using an object-oriented approach. Clearly, code quality is important. Your code should be easy to read and documented clearly. You have seen several examples in the course. The data structure is implemented in Python as a module, so that it can be used in a Jupyter notebook for demonstration purposes and testing, but also from a Python script for benchmarking on the HPC infrastructure. In case you prefer to work purely in a notebook, it will be up to you to figure out how to run that on the HPC infrastructure.
3. The implementation must be tested thoroughly for correctness as explained several times in the course.
4. Discuss the expected time and space complexity of your implementation.
5. The performance of the implementation must be tested as well using a large data sample. Time the insert and search functions for an increasing number of words and create plots. These benchmarks should be performed on the HPC infrastructure. Include the job script and the Python test script in your repository, as well as the output of the benchmark runs. Consult the VSC documentation if necessary.
6. Consider best case, average case, and worse case scenarios.
7. You can also benchmark a ternary search tree with a b-tree.

## Grading

- Code quality, correctness, and ease of use: 40 %.
- Testing and performance experiments, including HPC usage: 30 %.
- Experiment 6: 10 %.
- Use of version control & quality of commit messages: 20 %.

Note that if a team member has no or few commit messages, I must assume that person did not contribute to the project, so that person will receive 0 as a grade for the entire project.

# Questions

You will have occasion to ask questions on

- Q&A on Tuesday, March 27, 2025 at 17h00 CET
- Q&A TBD

You can also ask questions via email.

# Deadline

Deadline for the project: TBD (end of semester, either before or duing exam period).