# Exercise 4

# Parallel & Distributed Computer Systems

# Dec 26, 2019

Select one of the following two projects, or propose your own.

## 1. Parallel implementation of Vantage-Point tree (shared, GPU, distributed) and parallel all-kNN search, using the vantage-point structure

- Using the same interface as the one implemented in exercise 1, expand the vantage-point tree to:
  - Implement the vantage-point construction on GPU, using CUDA.
  - Compare execution time of construction with KD-trees for the same dataset.
  - Test with real datasets from UCI.
- Parallel all-kNN search using vantage-point tree

  Implement an exact parallel all-$k$NN search, using the vantage-point tree structure you implemented in the first exercise.
  - Report average number of points visited for computing the $k$NN of each point for different number of points and dimensions.
  - Test with real datasets from UCI.
  - **EXTRA**: Compare your implementation theoretically and experimentally with KD-trees (cite the implementation you used).

## 2. Sparse graph matrix reordering

Select one of the following algorithms for generating a permutation for sparse matrices and implement a parallel version using OpenMP or Cilk.

- `Dulmage–Mendelsohn`
- `Approximate Minimum Degree`
- `Nested Dissection`
- `Reverse Cuthill McKee`

Present results of the execution time required to compute the permutation. Compare the execution time of your implementation against existing implementation of the permutations.

## Deadline

Last exam day of February or September exam period (students can also submit in June, provided they are registered).

## Students that wish to suggest their own project, should contact the course instructors

## What to submit

- A 3-page report in PDF format (any pages after the 3rd one will not be taken into account). Report execution times of your implementations with respect to size $n$.
- Upload the source code on GitHub, BitBucket, Dropbox, Google Drive, etc. and add a link in your report.
- Check the validation of your code using the automated tester on e-learning.