# Minimum-Spanning-Tree (MST)

**Abstract**:

The goal of this project is to find optimized, parallelized implementations for the MST problem. Given a graph G = (V, E), we try to find a spanning tree T of G, with the sum of all edge weights of the edges in T being minimal. For simplicity, we will focus only on integer weights.

Our first goal is to create a synchronized and parallelized version of the Boruvka algorithm and compare the two of them. As a second step, we will implement additional MST algorithms and create a hybrid-algorithm, which finds the best MST implementation to call, based on graph criterias. The goal with this hybrid-algorithm is to outperform the parallelized Boruvka implementation.

For the implementation, we will try to look into various technologies, such as MPI, OpenMP and maybe even CUDA. To get suitable input graphs, we will implement random graph generators, creating graphs with different properties (e.g. sparse, dense, bipartite).

**References**:

<https://www.cs.ubc.ca/~condon/papers/chungcondon96.pdf>

**Team**:

Matteo Kamm ([matkamm@student.ethz.ch](mailto:matkamm@student.ethz.ch))

Simon Hrabec ([shrabec@student.ethz.ch](mailto:shrabec@student.ethz.ch))

Samuel Anzalone ([ansamuel@student.ethz.ch](mailto:ansamuel@student.ethz.ch))

Hulda Lilja Hannesdóttir ([hhannesdo@student.ethz.ch](mailto:hhannesdo@student.ethz.ch))

Mike Marti ([mikmarti@student.ethz.ch](mailto:mikmarti@student.ethz.ch))