

# Connected Components for Dynamic Graphs

## Bachelor's Thesis / Master Project

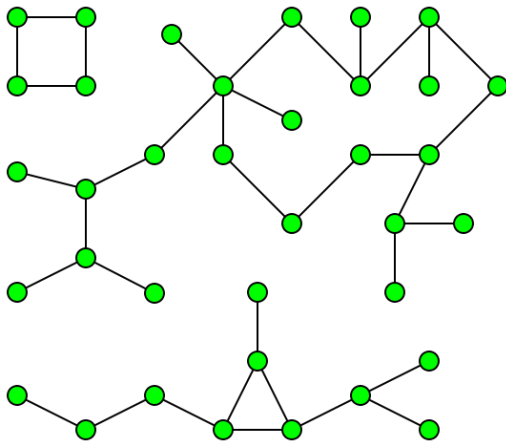


Figure 1: A graph with three connected components.

### Description:

We are currently developing a dynamic graph framework for GPUs, called *faimGraph*. As the pure representation of such a graph is only part of the appeal of graph processing, deriving interesting metrics from such a graph is typically the reason to manage a graph in the first place. The goal of this thesis is to implement *Connected Components* for *CSR*(Compressed Sparse Row) data structures as well as *faimGraph*. A connected component of an undirected graph is a *subgraph* in which any two vertices are connected by a path, and which is not connected to the *supergraph*. Such implementations typically have a lot of optimization potential that can be exploited as much as is permitted by the time frame of the project. Optionally, one could exploit the dynamic setting to avoid re-computation after a graph update.

### Objective:

- Implement basic version of *Connected Components* for CSR on the CPU
- Implement algorithm on the GPU for CSR and *faimGraph*
- (Optional) Exploit dynamic setting to avoid re-computation after a graph update

### Qualifications:

- Experience in C++
- Experience in CUDA, OpenCL or similar parallel computing framework
- Interest in massively-parallel compute architectures and graph processing

### Contact ICG:

Martin Winter  
 martin.winter@icg.tugraz.at