Project **Winter 2021/22**
Algorithm Engineering for hard problems **TU Berlin**
Nichterlein/Boehmer/Niedermeier **November 23, 2021**

3rd Exercise Sheet

Weighted Cluster Editing I

Solver: December 13 **Handout:** December 14 **Presentation:** December 15

Task 1. Data Reduction

Implement at least *six* data reduction rules presented in the lecture. Test the impact of applying the reduction rules with varying order and frequency. Submit the following programs.

(a) Submit your fastest search-tree algorithm (with or without data reduction rules). Your program's input and output has to be consistent with the provided benchmark script. In particular your program has to output the number of recursive steps.

(b) Submit a program which gets an input graph G (same format as in sheet 1), executes your implemented data reduction rules and outputs a potentially smaller graph G' (in the same format as the input) and an additional line `#weight:d` such that G has a cluster editing of weight k if and only if G' has a cluster editing of weight $k - d$.

(c) (optional) Submit a program which gets as input the output of your program from (b) followed by (after a line break) a cluster editing for the graph G' (in the same format your program in (a) has to output a solution). Let this cluster editing be of weight k'. The program should output a cluster editing of size $k' + d$ for graph G—the graph originally given to the program of (b). Please use lines starting with a # to pass additional information from your program in (b) to the program in (c).

Note: If you have a graph that has an edge with weight 0 then we say that this edge is not part of the graph but can be added without costs.

Task 2. Evaluation

Test your program from (a) on the provided instances using the benchmark script. Count the number of recursive steps by adding in the output the line "`#recursive steps:` x". The benchmark script will pass this information to the console.

Test you program from (b) on the provided instances. Try to measure the impact of each rule.

Hint: Further requirements on the evaluation are given in the next task.

Task 3. Presentation & Handout

Please submit a handout of your evaluation and present your findings with a short talk lasting at most 10 minutes.

The main focus of the handout should be the evaluation and comparison of different ways to apply the reduction rules. For these comparisons use your fastest implementation of the solver as baseline. As measures of impact you can use for example

- the number of recursive steps,
- the overall time, and
- the time per recursive step.

In addition to the one page limit for the text, there is an overall limit of two pages for your handout.

The main focus of the presentation should be a quick overview of your findings. In particular, show the most important changes you made for your fastest solver, and any great surprises on what works exceptionally well or not at all.

Differentiate between the instance types *random*, *real-world*, and *actionseq*. Come up with possible explanations for observed differences. Support your observations with quantitative statements.

Hint: You might want to adapt the benchmark script to test all these variants of your program at once. If you do not know how to adjust it, ask in the forum. We will then explain what to change.