
Exercise Sheet 7

Submit until Wednesday, June 27 at 4:00pm

Exercise 1 (12 points)

Implement the precomputation and query algorithm for Contraction Hierarchies. For the precomputation, contract the nodes in order of increasing edge difference using your contraction routine from the last exercise sheet. Initially, compute the edge difference for all nodes. Then, after each contraction, update the edge differences with one or more of the heuristics discussed in the lecture (lazy update, neighbours only, periodic update, spatial diversity).

You need not write a unit test for the precomputation (assuming you have already written one for the node contraction routine), but you should write one for the query algorithm (easy if you take the same graph as for the node contraction test).

As usual, consider the implementation advice given in the lecture and the code design suggestions linked from the Wiki. In particular, note that you can easily ignore arcs (in the precomputation as well as in the query algorithm) by using arc flags.

Exercise 2 (6 points)

Run the CH precomputation on one of our two datasets (preferably *BaWü*), and then execute 1000 random queries using the query algorithm explained in the lecture.

As usual, report your results in a row on the table linked from the Wiki. In particular, report the precomputation time, the total number of shortcuts added, the average query time, and the average shortest path cost.

Exercise 3 (2 points)

As usual, commit your code to our SVN and check that everything works on Jenkins, and also commit a text file *feedback-exercise-sheet-7.txt* where you briefly describe your experiences with this exercise sheet and the corresponding lecture.