

Exercise Sheet 1

Submit until Friday, May 20 at 2:00pm

Exercise 1 (1 point)

On the Wiki page of the course (see the URL in the header of this exercise sheet) you find the link to the registration at our course system Daphne. Please do that. That way you also get access to the SVN repository for this course.

Exercise 2 (7 points)

Write a simple class *RoadNetwork* for a directed graph with arc costs.

Add a method *readFromOsmFile* to your class that reads an OSM file (in XML format) and constructs the corresponding road network. To translate the road types to speeds, use the table that will be provided on the Wiki.

Add a method *computeShortestPath* to your class that computes the shortest path between two given nodes using the A^* algorithm. The heuristic values (a lower bound on the cost to the target for each node) should be an argument to that method.

Compute shortest paths on 1000 random queries using plain Dijkstra and using A^* with the straight-line heuristic, both on the OSM graph of Baden-Württemberg. Report the average running times for the two methods on the Wiki.

Note: For testing, you might want to start with a smaller graph first, e.g. the OSM graph of the Saarland. If your code takes too long on Baden-Württemberg, you may also report the average query times for the Saarland. But at least try Baden-Württemberg.

Exercise 3 (2 points)

Checkout a working copy of your folder in the SVN repository of the course (see your Daphne page for the URL). Add your code there. Also add a text file *feedback-exercise-sheet-1.txt* where you describe your experiences with the first exercise sheet (and, if you want, with the corresponding lecture). In particular, say how much time you invested and where you had problems and how much time that cost you. Commit everything to the SVN.