Graz University of Technology

**Institute of Geodesy**
**Working Group Navigation**

Ass. Prof. Dipl.-Ing. Dr. techn. Eva Buchmayer

Steyrergasse 30
A-8010 Graz

http://www.ifg.TUGraz.at

Tobias Hochreiter
tobias.hochreiter@student.tugraz.at

DVR: 008 1833 UID: ATU 574 77 929

# Navigation Systems
# Winter term 2025/26

## 1st Lab Programme
## *Routing*

### Objective

This program deals with computing the optimal routes to given points using Dijkstra's routing algorithm. The transport network of the city of Graz serves as data source. This network is given as an adjacency list with multiple costs.

The start node should be the actual nearest node to the home of one person in your group. Use the coordinates in the file *nodepl.txt* to determine which node shows the minimum coordinate difference to your home.

Starting from there, the optimal routes to three points within Graz must be computed:

- Basilika Mariatrost (node 9328)
- Schloss Eggenberg (node 6031)
- Shopping-Center Murpark (node 8543)

It is possible to assign all sorts of costs for the cost function to calculate the optimal path. For the three defined routes, two different costs - **time** and **distance** - should be evaluated.

In addition to this task, choose one of the two following tasks:

2.1) Develop **your own cost-function** and apply it to the routing algorithm. You are free to use all columns of the arclist for your self-developed cost function as well as external data. Compare the calculated paths of the different cost functions.

2.2) In addition to the Dijkstra algorithm implement the **heuristic algorithm A*** and calculate the 3 routes with this algorithm. Use either **time or distance** as cost within the A* algorithm (it is sufficient if you choose one of the two to calculate the optimal paths with A*). Finally, compare the results of the Dijkstra and A* algorithms.

### Data source

The transport network of Graz is given as an adjacency list within three ASCII text files:

- *nodepl.txt* contains the coordinates of the nodes. The row numbers correspond to the node ID, the values are the coordinates ($\varphi$ and $\lambda$ in [°]).
- *nodelist.txt* is the adjacency list of nodes.

  The row numbers correspond to the node IDs, the values are the outgoing valences of these nodes. Outgoing valences display the number of outgoing arcs.
- *arclist.txt* is the adjacency list of the arcs.

  The rows correspond to the output valences of the adjacency list of nodes. The first column includes the destination nodes. In the following, the different costs per arc are given:
  - 2nd column: time [h]
  - 3rd column: distance [km]
  - 4th column: speed limit [km/h]

- 5th column: clazz
- 6th column: flags

The time is derived from the quotient of the distance and the speed limit on the specific arc. It is given in the form of a directed graph (e.g., for one-way streets, the time is set to 100000 h). The clazz and flags correspond to arc classifications. The classification for clazz is given in table 1, the classification for flags is given in table 2.

**Table 1: clazz classification**

| Clazz number | Type of street |
|---|---|
| 11, 12 | Highway |
| 15, 16 | Primary way |
| 21, 22 | Secondary way |
| 31 | Tertiary way |
| 41 | Residential way |
| 43 | unclassified |

**Table 2: flags classification**

| Flags number | Type of use |
|---|---|
| 1 | Car |
| 2 | Bike |
| 3 | Car and bike |
| 4 | Pedestrian |
| 5 | Car and pedestrian |
| 6 | Bike and pedestrian |
| 7 | Car, bike and pedestrian |

## Visualization

Note that for visualization, you must store your list of predecessors. Do not forget to label your map correctly with appropriate units. Graphs and figures must have proper scaling, labelling and linewidths, too.

## Deliverables

- Visualization of the optimal routes with time and distance as costs.
- Comparison of these routes.
- If you have chosen task 2.1):
    - Creation of a meaningful, self-developed cost function.
    - Visualization of the optimal routes with the self-developed cost function in the map.
    - Evaluation and interpretation of the different cost functions and the process of the Dijkstra algorithm.
- If you have chosen task 2.2):
    - Visualization of the routes calculated by both algorithms (A* and Dijkstra) in a map.
    - Comparison and interpretation of the results of both algorithms and explanation of the implementation of A*.
- Source code of the developed software, well formatted with explanatory comments (uploaded to the TeachCenter).
- Prepare a **presentation with audio commentary** (e.g. with PowerPoint) and save it in the format **.mp4**. The length of the video should not exceed 5 minutes and should include your methodology, results and especially your results with, depending on your choice, the self-made cost function or A*.
- Try to show with your presentation that you have understood and implemented the given task and
    - (for 2.1) persuade your viewers that your self-made cost function is the most efficient for your needs or
    - (for 2.2) explain to your viewers the pros and cons of A* over Dijkstra.

## Presentation / Handing in the lab programme

You need to be in a group to submit the lab programme. The source code and your video of the presentation must be uploaded to the TeachCenter by November 16th, 11:59 p.m.

For **submission of the source code**, please create a ZIP-folder containing:

- The source code, named as follows: *lab01_ws2025_26_GROUP#_SURNAME1_SURNAME2.py* (the file extension may vary, depending on the used programming language). If your code consists of multiple files, add a suffix to the file names (e.g. *lab01_ws2025_26_ GROUP#_SURNAME1_SURNAME2_main.py*).
- A short **README-File** (*README.txt*), containing additional information (e.g. about the approximate runtime of the program, errors/exceptions that may be raised, the version of the used programming language etc.)
- Additionally, the folder should contain all the data which is necessary to run the code.

For the **submission of the video**, please name the video according to your group name, e.g. *GroupA.mp4*.

Eva Buchmayer and Tobias Hochreiter, 20 October 2025