

# Algorithms and data structures – project description

Wintersemester 2024/2025  
Prof. Dr. Barbara Sprick

November 11, 2024

# Disaster management planning tool for the city of Schilda

The formerly small town of Schilda has now grown into a big city. In order to be able support the citizens even in the event of a disaster, the city is now revising its disaster management plan. This also includes the development of a planning tool that will help to react flexibly to different disasters and the resulting challenges. The city of Schilda requests you to develop this disaster management tool

The individual functions that are required can be found on the following pages.

The disaster management tool should provide the following functionality:

1. B1: Load and display city map
2. B2: Modify city map (enter road closures and damage locations)
3. F1: Rebuild communication infrastructure
4. F2: Plan evacuation routes
5. F3: Plan routes for emergency services
6. F4: Set up supply points for emergency services
7. F5: Plan deployment of emergency services

# B1: Read city map, output

## Basic function

This function of the planning tool shall read a city map in the form of an adjacency matrix and output it.

You can find a test example of an adjacency matrix in Moodle.

## B2: Modify city map

### Basic function

This function shall be used to add important points to the city map, such as deployment sites, assembly points, supply points, etc., so that these can be taken into account in the following planning tasks.

Note: supply points are not always located at intersections of the existing road network. Therefore, the distance to the next accessible intersections should also be indicated.

In addition, this function should offer the possibility to mark roads that have become impassable as such.

Additional option: In some cases, roads may be flooded and only accessible by boat. Expand your city map so that you can distinguish between roads (land routes) and waterways and take these into account in your calculations.

# F1: Rebuilding the communication infrastructure

In the event of a disaster, the communication infrastructure such as radio masts, fiber optic networks, etc. may be destroyed. Here it is important to first create a basic network that allows the most important nodes, such as rescue stations, hospitals, and government buildings, to communicate with each other reliably and stably.

This function should calculate such a basic plan, taking into account minimal repair costs and resources. In the first step, direct lines (radio relay, field cable, etc.) should be realized.

Input: A plan of possible infrastructure nodes with repair costs for the route to the surrounding nodes.

## F2: Evacuation Routes

In an emergency, it may be important to evacuate citizens from a danger zone as quickly as possible. This functionality of the tool is designed to take into account the current situation (collection points for the people to be evacuated, roads that have become impassable, or the capacity of available infrastructure such as buses, etc.) and calculate whether the existing infrastructure is sufficient to evacuate all people from the collection points to safe emergency shelters or whether additional infrastructure is needed. Additional option: Additional consideration of routes that are only accessible by boat across water. For this, resources such as boats or combined options should also be considered. Input: List of assembly points and emergency shelters, updated city map with capacities for the individual routes.

## F3: Route planning for the emergency services

In the event of a disaster, it is important that emergency services arrive at the scene as quickly as possible. Develop a navigation system that calculates the fastest route to the scene for the emergency services, taking into account roads that may have become impassable.

Additional option: It may be that the scene of the emergency can only be reached by boat. The planning tool should provide an extension that calculates the fastest hybrid route (car + boat) for such scenes of emergency as well.



## F4: Supply of emergency services

The relief forces also need to be supplied regularly. Since the operation sites are spread all over the city, the disaster control sets up supply points for relief forces. This function is designed to help determine the optimal locations for  $k$  additional supply points.

The aim is to minimize the average distance of all relief forces to the nearest supply point and to distribute the relief forces as evenly as possible among these points.

Input: city map with deployment locations, possibly with waterways

## F5: Deployment planning for emergency services

New emergency services personnel with **specific skills** are urgently needed at the individual deployment locations. Emergency services personnel who are not currently assigned to a specific task are located in the staging area.

Emergency services typically work together in groups or squads. They are also assigned certain resources, such as wheel loaders, pumps or chainsaws. This function of the planning tool is designed to enable an efficient distribution of the emergency services available in the staging area to the deployment sites. In doing so, the required resources and skills are to be compared with those available in each case.

Additional option: Sometimes it makes sense to first supply an operation site with all the equipment it needs, while a second operation site has to wait. In another case, it may make more sense to supply both operation sites as much as possible.

# Task description

1. Develop a graph data structure that allows the representation, input, output and modification of the city map as an adjacency matrix. (Functions 1 and 2) (Basic)
2. Implement the underlying graph algorithms for tasks. Use the provided text file (adjacency matrix) as test data (Basic)
3. Implement the functionality described above using the implemented graph algorithms and the implemented input/output and graph modification. (You may need additional data structures. Please implement these appropriately as well.)
4. Programming language: Java or Python

# Submission

Each team should submit

- ▶ The commented program code (in Gitlab)
- ▶ Documentation with the following content:
  - ▶ For each function, please describe how you model the problem, which graph algorithm and which data structures you use.
  - ▶ Describe your input for the respective function.  
(Example communication infrastructure: A graph in the form of an adjacency matrix is read in. The nodes correspond to the possible locations of the nodes for communication, the edges correspond to the possible connections between the nodes, each with repair costs. ) Please describe all data structures used and explain why you have chosen this data structure for the respective application. A complexity analysis of your implemented algorithms.