Exercise Sheet 2

Exercise 1

Consider the following

Problem
A group of 3 soldiers (Team-A) captured another group of 3 soldiers (Team-B).
Team-A must bring Team-B to their camp but for doing so they have to cross a river.
The river can be crossed with a boat that can contain only 2 people at most (both from Team-A, Team-B, one for each Team, or nobody).
If the number of Team-B members at any time is larger than the one of Team-A on one of the two sides of the river, Team-A can be overtaken, therefore this should not happen [1].
Find a sequence of moves which bring safely both Teams on the other side of the river.

- Represent the states with a triple (x,y,z), where $x, y \in \{0, 1, 2, 3\}$ and $z \in \{0, 1\}$. x(y) represents the number of soldiers in Team-A(B) on the initial side of the river. z represents the location of the boat (1=initial side, 0=other side).
- The initial state is (3,3,0).
- The goal state is (0,0,0).
- Solve the problem implementing a **breadth-first search** respecting the constraints of the problem.
- If possible, favour an object-oriented programming approach, making your breadth-first search alfgorithm independent from the specific data structure of the problem.

Exercise 2

- What is the state-space size?
- How many states contains the path to the solution your BFS algorithm found? Report one solution **and your code**.

Due date: Wednesday, May 8th 2024

[1] A situation where the #Team-A = #Team-B is therefore allowed.