Assignment on Constraint Programming

Prof. Dr. Marc Pouly
Lucerne University of Applied Sciences and Arts

This assignments consists of modelling and programming exercises with OR-Tools in Python and Jupyter Notebooks. All the following puzzles have been introduced in the lecture. Look at the slides again before you start. Exercise 1 to 4 can be solved after the first lecture on constraint programming already. Exercise 5 requires lecture 4 on routing problems. The puzzles in exercise 1 to 4 have exactly one solution by construction. When your model finds more than one solution, it cannot be correct ☺

Exercise 1: Sum Frame Sudoku (1 Point)

Write a constraint model for the 9x9 Sum Frame Sudoku.

Exercise 2: Magic Square (1 Point)

Write a constraint model for magic squares of user-defined size.

Exercise 3: Binoxxo (1 Point)

Write a constraint model for the 10x10 Binoxxo puzzle.

Exercise 4: Xmas Puzzle (1 Point)

Write a constraint model for our Xmas puzzle.

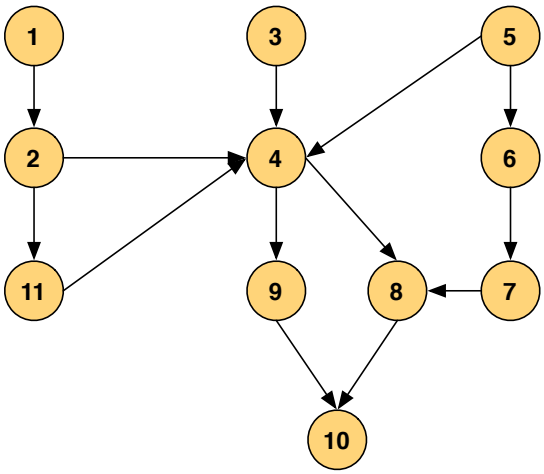
Exercise 5: Routing (1 Point)

The following exercise originates from a research project at HSLU.

Our industry partner is a Swiss small to medium sized company in the electricity market. They have one central headquarter from where they send out teams for various repair and maintenance tasks. Every task is bound to a specific location. The company wants to minimize the overall travelling costs. For simplicity we assume that the costs between any pair of locations are known in advance. However, they additionally have to respect certain inter-dependencies among the tasks. For example: for safety reasons it is important to first switch off electricity (task A) before starting the repair work on electric components (task B). This induces a constraint that task A must always be completed before task B can be started.

Write a constraint optimization model based on the OR-Tools Routing API that takes the number of teams as input together with a costs matrix between any pair of locations. For testing we provide code that automatically creates cost

matrices with random numbers. The following figure displays the inter-dependencies among 11 tasks to be worked off. We further assume that location 0 (not displayed here) refers to the company's headquarter. The duration for each tasks can be considered a constant and can thus be neglected in your model.



Hint: The slides contain all necessary code snippets.