

EXERCISE 1

Exercise 1 Route searching

1. Create a set of cities (as points) with coordinates x, y on a plane with height as z coordinate. The cost of going from city A to city B is equal to the Euclidean distance between two cities, if there exists a road. You should define scenarios according to two criteria:

1. There are all the direct connections / c.a. 80% of possible connections
2. The problem is symmetrical / asymmetrical (in asymmetrical – going up is height +10%, going down: -10%)

You should choose the coordinates randomly from the range $<-100, 100>$ for x, y and $<0, 50>$ for z .

2. Represent the created map as a weighted (directed) graph, where cities are the nodes and roads are the edges of the graph.

3. In the created scene, solve the traveling salesman problem: The salesman starts from a chosen city and has to visit every city exactly once before returning to the starting city. The goal is to find a path with the lowest cost.

In the problem, we define state as a partial or full path from the starting city and the corresponding state. You should represent the search problem in a form of state tree.

1. Implement a full search of the tree, using BFS and DFS methods.