FSD301: MST with Prim's algorithm

Author : Christophe Garion
Audience : SUPAERO 3A SD
Date :

1 Presentation

The main objective of this session is to implement Prim's algorithm to compute Minimum Spanning Trees. The lab is language agnostic, i.e., you can choose whatever programming language you prefer to implement it.
Prim's algorithm is a greedy algorithm: it solves the problem by locally optimal choices at each stage.

2 Questions

1. define your own data structures to represent nodes and set of nodes, as set of nodes are needed in the algorithm. You may of course use the networkx Python library.
2. implement Prim's algorithm and test it on the example presented during lecture.
3. test your implementation of Prim's algorithm on the problem contained in the `edges.txt` file.

   The first line in `edges.txt` gives you the number of vertices and the number of edges in the graph. Each following line represents an edge with the source vertex, the destination vertex and the cost of the edge. A Python script `reader_students.py` is available to help you understand how to extract the vertices from the files.

   The sum of the cost of the edges in the MST should be `-3612829`.
4. (advanced) write a better implementation of Prim's algorithm using a binary heap. You may reuse Python standard implementation of a heap. Read the documentation to understand how to deal with updates of nodes.

   Verify that execution time is better than the naive implementation.

3 Documents to be uploaded

You should upload your solution in the corresponding LMS repository. Either upload a single file or an archive, but do not forget to **add your names in the file name**.

License CC BY-NC-SA 3.0