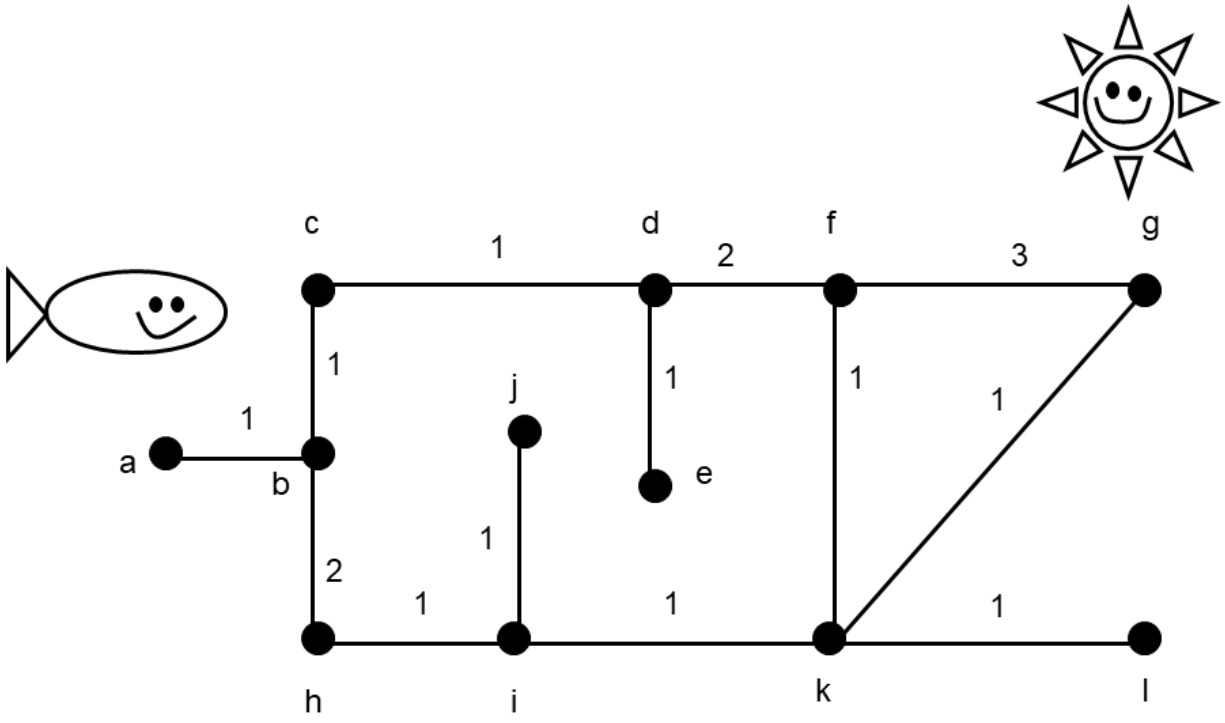
PROBLEM SET 3: Graphs and Trees **EDyL 2019-2020**

[Publication date: 2019/10/29]
[Due date: 2019/11/07, 09:00]
[Solutions (in class): 2019/11/07]

NOTE: Make sure to include explanations in your answers. An exercise whose solution is correct but does not include an explanation can be considered incomplete and not receive full credit.

NOTE: If there are different alternatives in any of the steps of the algorithms, use alphabetical ordering to decide which alternative is selected. Use the convention that digits precede letters.



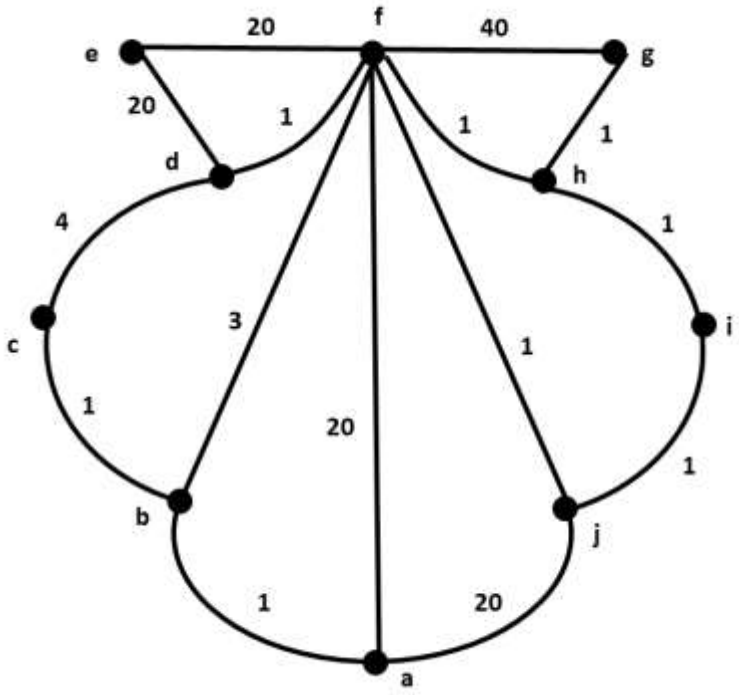
This graph will be used in exercises 1 to 3

EXERCISE 1: Help Little Fish find its way out of the labyrinth! To this end, find the path from **node a** to **node q**, making use of the Dijkstra's algorithm. Detail the steps of the algorithm in the table below, and indicate the optimal path and its weight. Use as many columns and rows in the table as necessary.

EXERCISE 2: For maintenance labor in the labyrinth, it is necessary to obtain the breath-first-search tree corresponding to the graph, starting from **node a**. Represent the tree and explain the procedure to obtain it.

EXERCISE 3: We wish to implement a way to access all the nodes of the labyrinth **from node a** at the minimum cost possible. We also wish to preserve the connectivity of the selected subgraph at each stage of the process.
If several alternatives are possible at some step of the application of the algorithm use alphabetical ordering to determine the order of exploration. Employ as many rows in the table as needed. If at some point an edge is discarded, indicate the reason why it is discarded. Represent the resulting tree, indicating the cost of the identified access. Give the name of the algorithm, if it is a known one.

Edge considered	Weight	Select/Discard



This graph will be used in exercises 4 to 6

EXERCISE 4: For maintenance labor on the *camino* to *Santiago de Compostela*, it is necessary to obtain the depth-first-search tree corresponding to the shell graph, starting from **node f**. Represent the tree and explain the procedure to obtain it.

EXERCISE 5: Find the optimal (lowest cost) path to *Santiago de Compostela* from **node a** to **node q**, making use of Dijkstra's algorithm. Detail each of the steps of the algorithm in the table below, and indicate the optimal path and its weight. Use as many columns and rows in the table as necessary.

EXERCISE 6: We wish to implement a way to access all the nodes of the shell at the minimum cost possible. At each step, the cheapest among the remaining links is chosen. If several alternatives are possible at some step of the application of the algorithm use alphabetical ordering. Employ as many rows in the table as needed. If at some point an edge is discarded, indicate the reason why it is discarded. Represent the resulting tree, indicating the cost of the identified access. Give the name of the algorithm, if it is a known one.

Edge considered	Weight	Select/Discard