

MATH.APP.270 Algorithms for graphs

Programming assignment 2

2023

This programming assignment is concerned with flow networks.

The starting point for the maximum flow problem is a flow network (V, E, c) and a source vertex $s \in V$ and a sink vertex $t \in V$. The goal in the maximum flow problem is to find a *flow* which is as large as possible. A flow is specified by the triplet (s, t, f) , where f is the flow function. Details on the maximum flow problem can be found in the course notes.

The course notes contain an outline of the the Ford-Fulkerson algorithm that solves the maximum flow problem. Your task in this assignment is to write code that implements the Ford-Fulkerson algorithm. Your code should fulfill the following requirements:

- The input to your code should be a weighted graph (V, E, c) , a source vertex s and a sink vertex t . The format of a file containing a weighted graph is given on the Moodle page.
- Your code should produce a flow function f which solves the maximum flow problem. If the maximum flow problem has no solution, your flow function should contain zero for all flows in the network.
- Your code should contain a counter, that counts the total number of edges processed when producing the solution to the maximum flow problem. An edge is processed when it is tested when searching for an augmenting path. (Note: the same edge may be processed several times).
- The Ford-Fulkerson algorithm presented in the course notes uses a procedure called AUG for finding an augmenting path flow. You will have to write code to compute an augmenting path.
(Hint: it is suggested that you use a BFS-approach.)

If your code is written in Python, then the flow function produced by your code should be in a dictionary data structure. The Python code to initialize flow f as a dictionary is as follows:

$$f = \{\}$$

Assuming an edge (u, v) exists, then a flow of h along such an edge can be assigned as follows:

$$f[(u, v)] = h$$

The flow function produced by your code must satisfy the conditions given at the start of Section 5.1 in the course notes.

You may use some language other than Python, but if you do you must also meet the following requirements:

- Your code should accept as an input a weighted graph that is stored in the form specified on the course Moodle page.
- Your code must also accept a source vertex s and a sink vertex t .
- You must provide clear and detailed instructions on how to compile your code and how to run your code. You cannot assume that a user has any particular knowledge. For example, you should not assume that user knows anything about Java or how to use Netbeans.