Practices of Discrete Mathematics: Introduction to Graph Theory

Sesión 9 (Maximum flow)

Problem

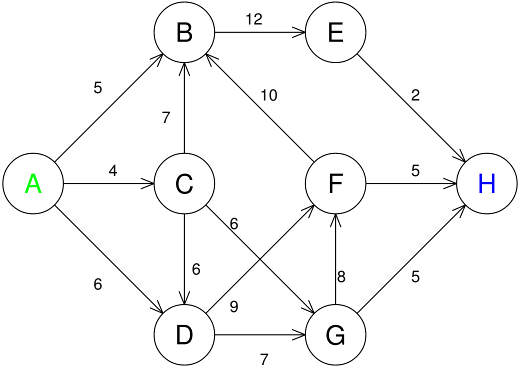
An agent of a travel agency must arrange a travel for 12 tourist from Madrid to Saint Petersburg in a certain date. He knows the number of free seats of each useful flight to travel.[a] These are detailed in the following table:

Flight	Free seats
Madrid-Paris	5
Madrid-Frankfurt	4
Madrid-Brussels	6
Paris-Helsinki	12
Frankfurt-Paris	7
Frankfurt-Brussels	6
Frankfurt-Warsaw	6
Brussels-Riga	9
Brussels-Warsaw	7
Helsinki-St. Petersburg	2
Riga-Paris	10
Riga-St. Petersburg	5
Warsaw-Riga	8
Warsaw-St. Petersburg	5

Is it possible to arrange the travel for all tourists? If the answer is yes, how to do it?

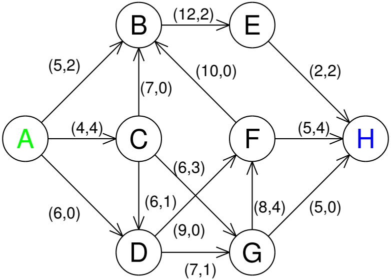
[a] It is assumed that all connections are possible

The problem can be represented by means of a weighted directed graph whose vertices correspond with the cities and whose edges correspond with the flights. The weights (**capacities**, in this context) are the numbers of free seats.



The correspondence between vertices and cities is the following one: A: Madrid, B: Paris, C: Frankfurt, D: Brussels, E: Helsinki, F: Riga, G: Warsaw, H: Saint Petersburg.

A **flow** is any possible set of weights (assigned to the edges) that represent the number of tourists in each flight. A possible example of flow is the following one: 2 travelers follow the path A-B-E-H, 4 of them go from A to C and, when they are in C, there is a bifurcation: 3 of them follow the path C-G-F-H and the remaining one C-D-G-F-H. We can represent this flow (jointly the capacities) in the following way:



Notice that this flow only allows to travel to 6 tourists.

Objective: to find a flow that allows to travel to the maximum number of tourists.

Network

A network is a weakly connected weighted directed graph without loops and with two special vertices:

- a source, with out-degree > 0 and in-degree $= 0$,
- a sink with in-degree > 0 and out-degree $= 0$,

and such that the weights are non-negative integers. The weight of an edge e is called capacity of e and it is denoted by $c(e)$.

In our example, we have a network with a source at the vertex A and a sink at the vertex H, and with the indicated capacities.

Flow

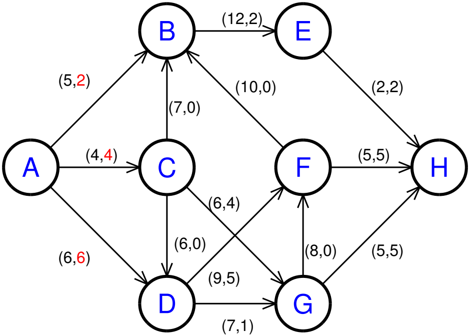
A flow f of a network is a collection of weights (different from the one of capacities) called flows such that:

- The weights (flows) are non-negative integers.
- The flow of each edge e (denoted by $f(e)$) is less than or equal to the capacity of e, that is, $f(e) \leq c(e)$.
- For each vertex v that is neither the source nor the sink the sum of the flows into v is equal to the sum of the flows out from v.

Value of a flow

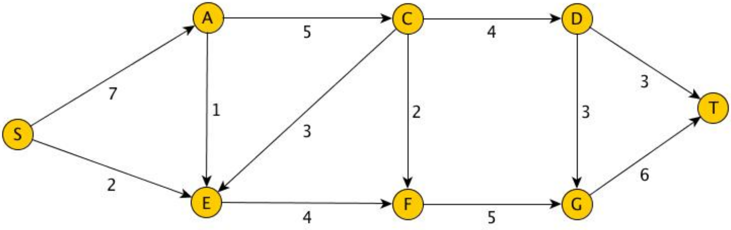
The value of a flow f, denoted by $v(f)$, is the sum of the flows out from the source.

There exists an algorithm (Ford-Fulkerson algorithm) that computes a maximum flow of a network. Applying this algorithm it is obtained, as maximum flow, the following one (whose value is 12):



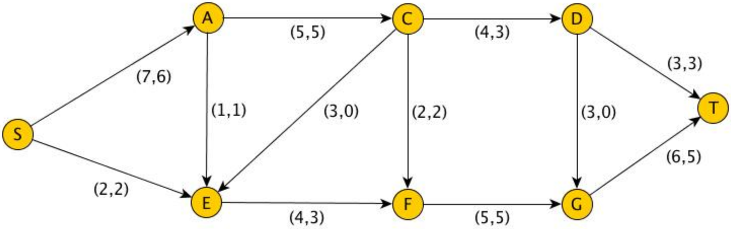
Example

In the following example we show a network with the associated capacities:



Example

Applying the Ford-Fulkerson algorithm we obtain the following maximum flow:



Its value is 8.

Problem

Suppose that we have the following list of chemical products and its corresponding amount of containers.

$$A(7), B(6), C(5), D(6), E(10)$$

We want to transport them from your warehouse to another one. For this task we have 4 lorries whose capacities in containers are the following:

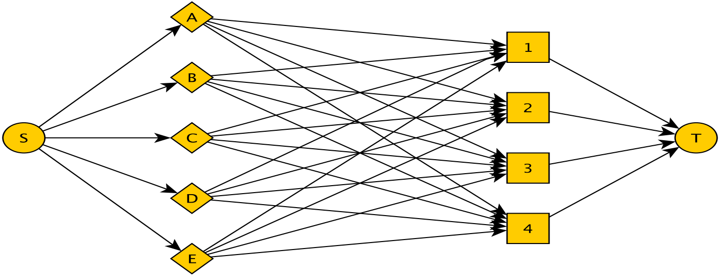
Lorry 1 (4), Lorry 2 (7), Lorry 3 (6), Lorry 4 (5)

Border rules impose that you cannot put more than 2 containers of the same type in each lorry.

How many containers can you transport and how would you arrange them?

Modeling the problem

This problem can be represented by using the following network:



Question:

Which are the capacities that we must assign to the edges in order to solve the problem using the Ford-Fulkerson algorithm?