Complex-Network Modelling and Inference

Lecture 5: Nodes, paths and cycles

Matthew Roughan
<matthew.roughan@adelaide.edu.au>
https://roughan.info/notes/Network_Modelling/

School of Mathematical Sciences,
University of Adelaide

March 7, 2024

Section 1

Neighbourhoods, Node Degree and Eulerian Cycles

Neighbourhood and node degree

Definition

The *neighbourhood* of node i is defined by

$$N_i = \{j \mid (i,j) \in E\},$$

i.e., the set of nodes adjacent to i.

Definition

The *node degree* k_i is the number of neighbours of node i, *i.e.*,

$$k_i = |N_i|.$$

Definition (Alternative definition)

The *node degree* k_i is the number of edges incident to i, *i.e.*,

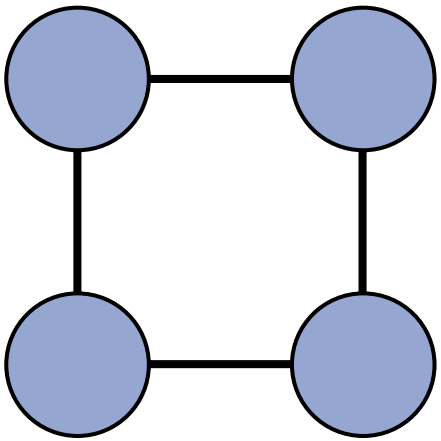
$$k_i = \left|\{(i,j)|(i,j) \in E\}\right|.$$

Regular graphs

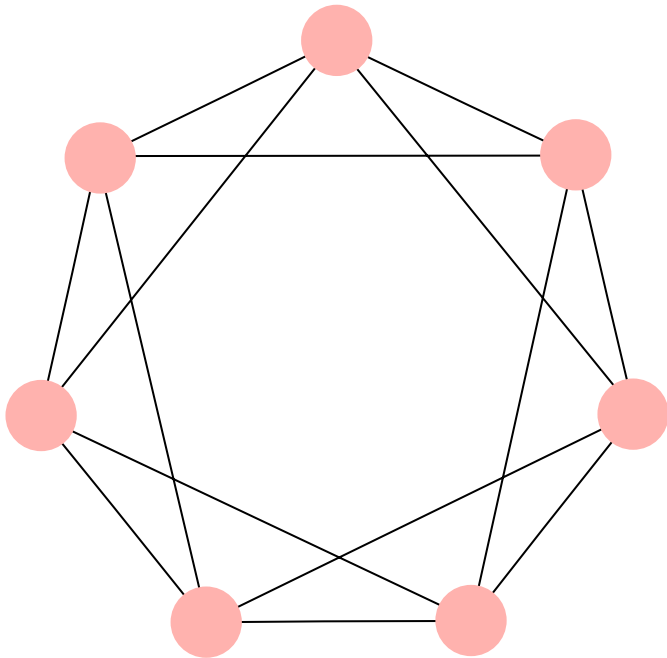
Definition

A network is *r-regular* (or just regular) if every node has the same degree, *i.e.*, $k_i = r$ for all i.

2-regular



4-regular



Can I have a 3-regular graph?

Node degree facts

Theorem (Handshake theorem)

For any undirected graph

$$\sum_i k_i = 2|E|.$$

Proof.

(1) The node degrees are given by a column or row sum of the adjacency matrix A.
(2) The number of edges in an undirected graph is the sum over all elements in the upper (or lower) triangle of A, and A is symmetric, so the sum over all A is $2|E|$. □

Intuitively, each edge contributes to the degree of the node at either end, and hence adds 2 to the total of all the degrees.
The result goes back to Euler (1736).

Node degree facts

Corollary

The number of odd degree nodes is even.

Proof.

The sum of the degrees is $2|E|$, which is even. The only way to get an even total would be if there is an even number of odd components to the sum. □

Node degree for directed graphs

- a node's **in-degree** is the number of links connected to it, i.e.,

$$\text{in-degree}(i) = \Big|\{(j,i)|(j,i) \in E\}\Big|$$

- a node's **out-degree** is the number of links connected from it, i.e.,

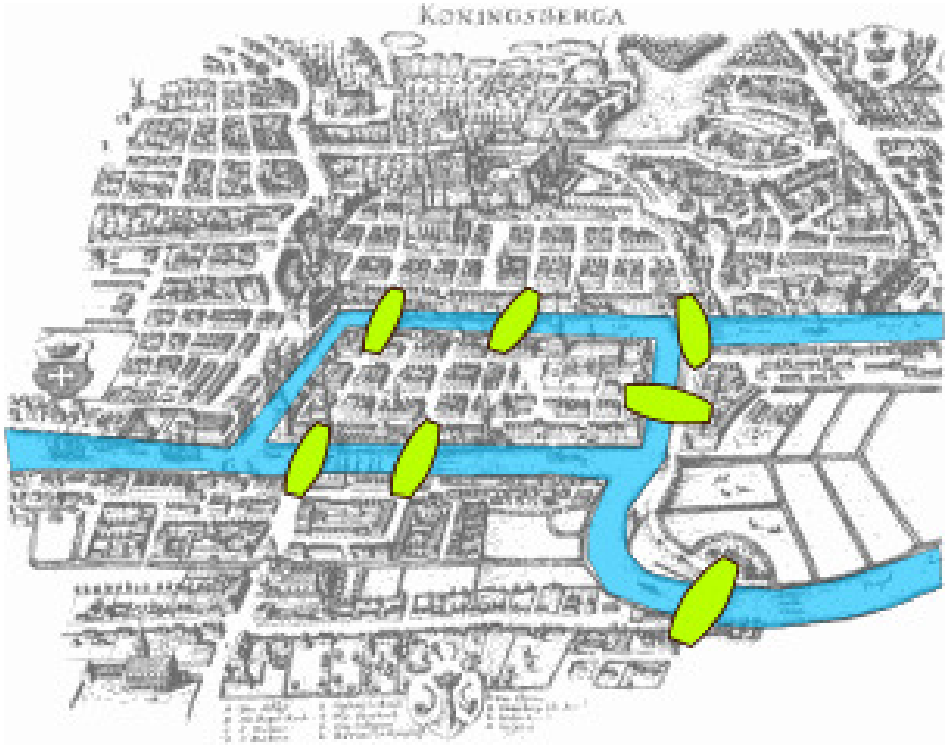
$$\text{out-degree}(i) = \Big|\{(i,j)|(i,j) \in E\}\Big|$$

The node in-degrees are given by a column sum of the adjacency matrix A, and out-degrees by a row sum.

Königsberg Bridge problem

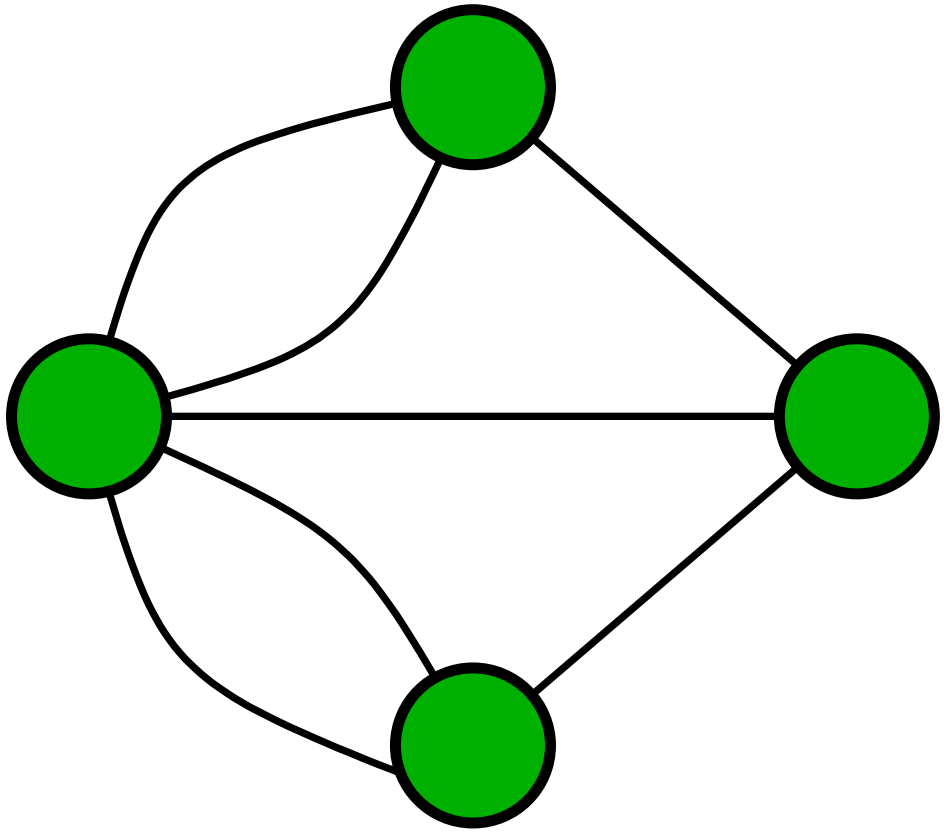
Why do we care about node degree?

- First graph theory problem (by Euler, 1736)
- Find a (walking) path that crosses all the bridges in Könisgberg exactly once.



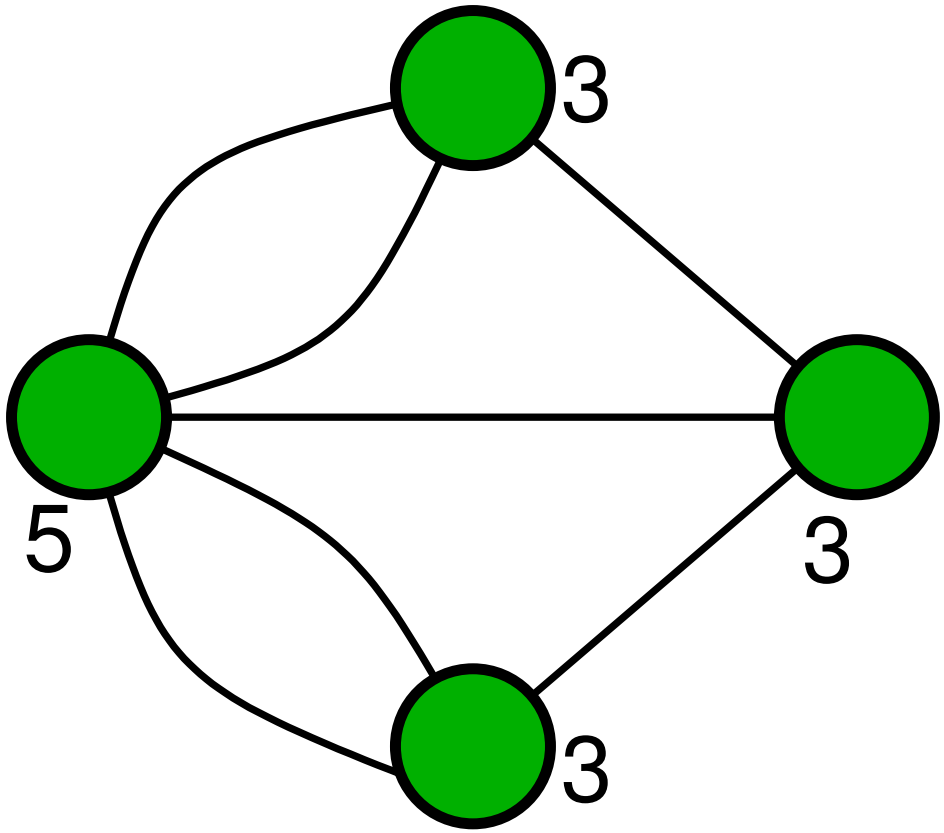
Königsberg Bridge problem

- Each land mass is a vertex
- Bridges connect land masses



Königsberg Bridge problem

- Node degree is critical
- End points must have odd degree, others need even degree (because we enter and then leave)



Eulerian cycles

Definition

An *Eulerian path* is a path that crosses every edge exactly once. An *Eulerian cycle* also returns to its initial node.

Theorem (Euler, 1736)

An Eulerian cycle exists iff the graph is connected, and the node degrees are all even.

Proof.

$\Rightarrow$ (necessity)
Such a cycle cannot exist (trivially) if the graph is not connected.
If the degree of any node is odd, then one can enter that node from some path, but not leave it (after some number of passes), and hence such a cycle cannot exist. $\square$

Eulerian cycles

Proof.

$\Leftarrow$ (sufficiency)
Start at any node a, and form a path by choosing any link from a, and so on. As the node degrees are even you will always be able to leave a node you enter, so you will eventually return to a.
There are two possibilities:

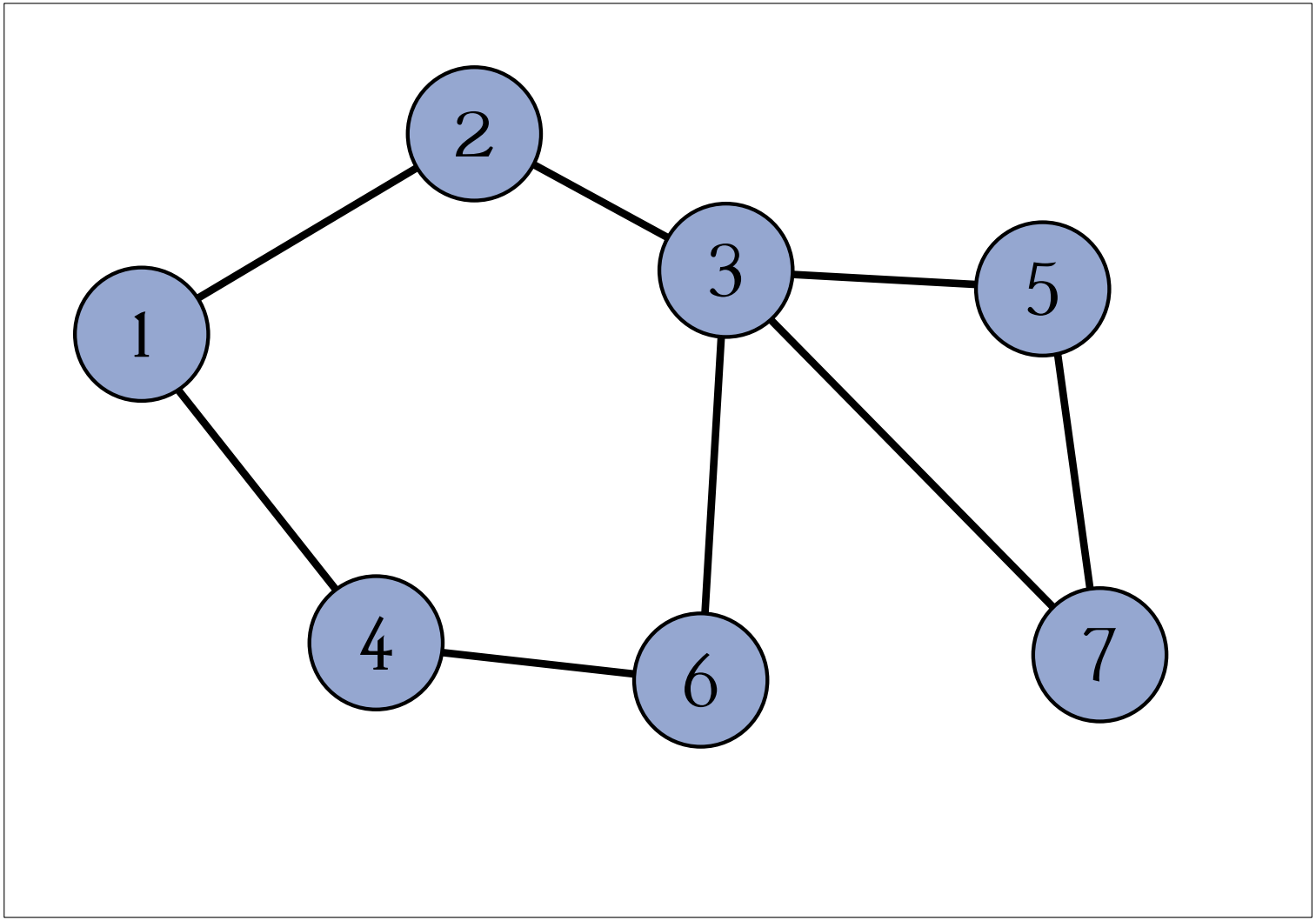
1. You have constructed the Eulerian cycle. Yay!
2. There are some edges missing from the cycle.

In the latter case, there must be at least one node b on the existing cycle with unexplored edges. Otherwise, there are nodes that are unconnected to the existing path.
Start a new cycle at b as before. When finished stitch it to the old path to form a continuous cycle.
Then repeat this procedure until you have an Eulerian cycle. Each step reduces the number of extant edges, so it must converge. □

Finding a Hamiltonian cycle is more challenging.

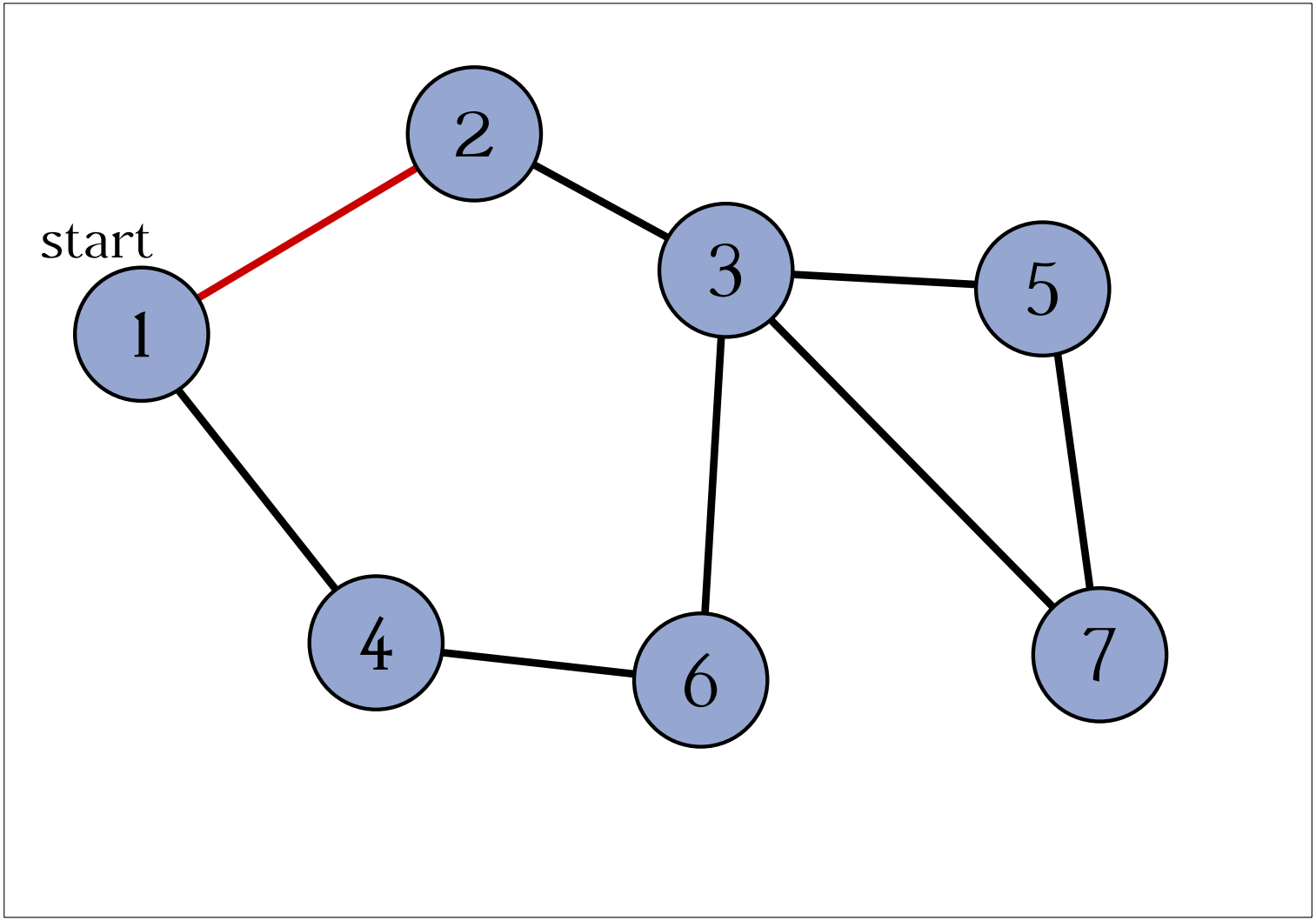
Eulerian cycles

Example:



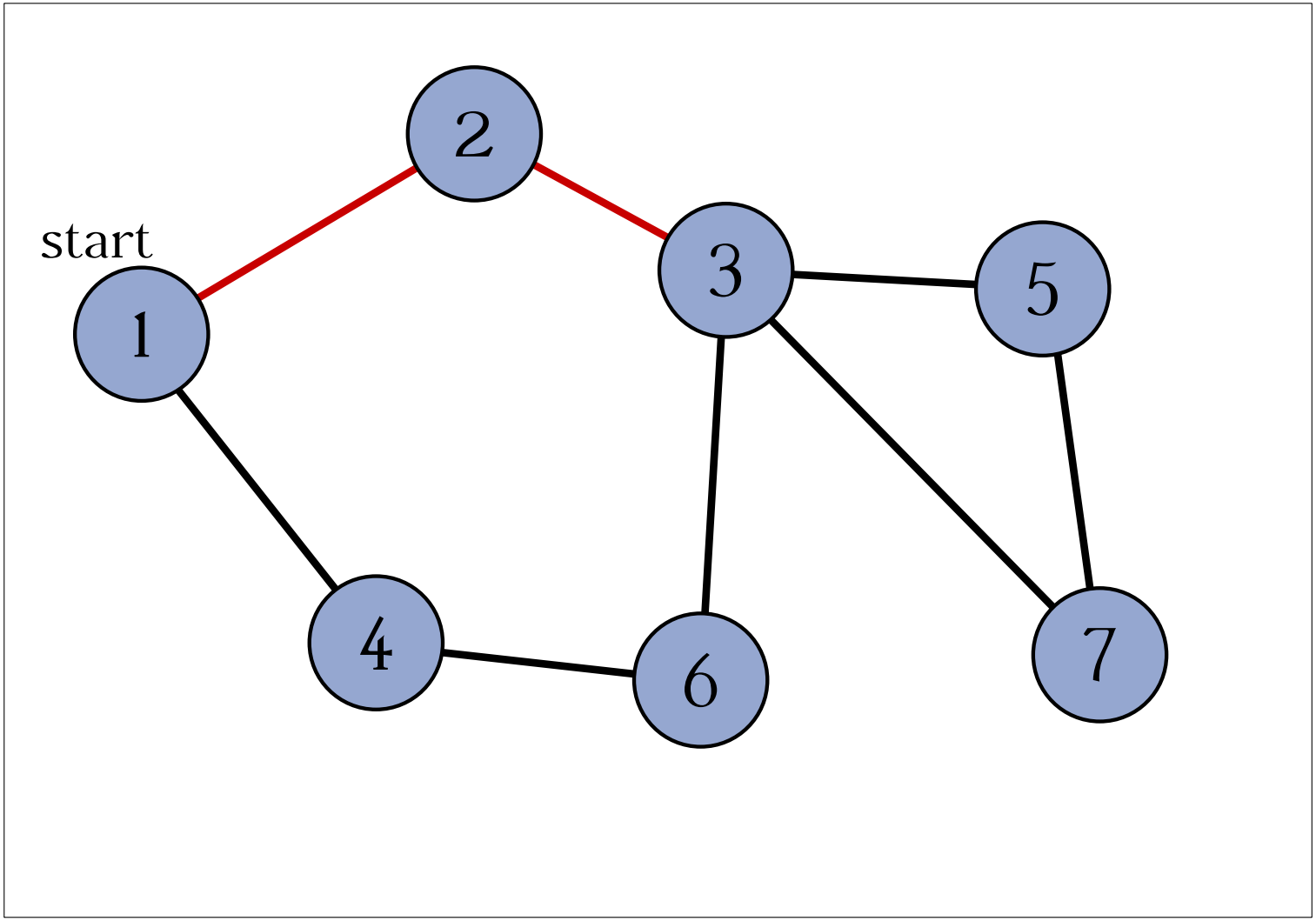
Eulerian cycles

Example:



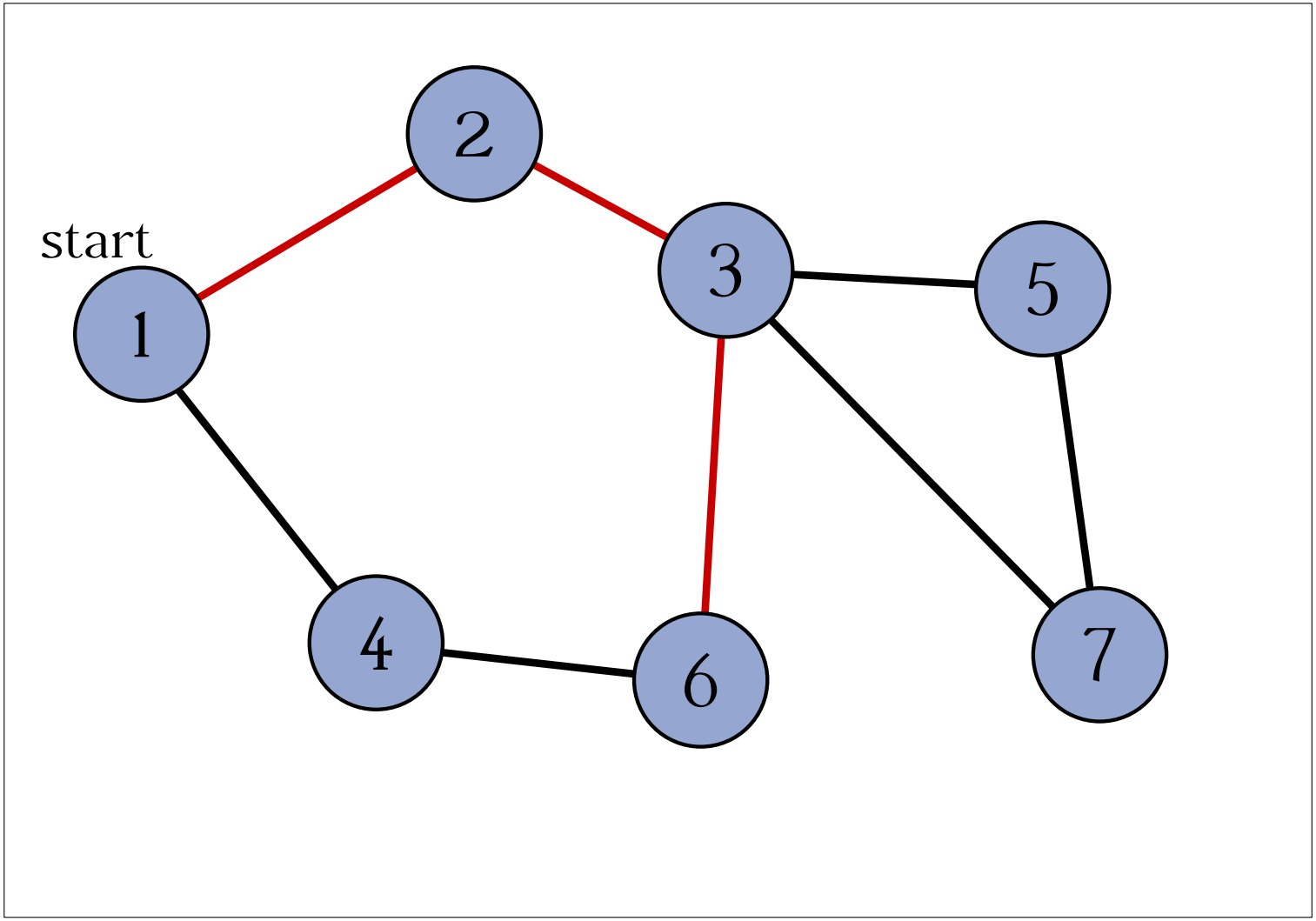
Eulerian cycles

Example:



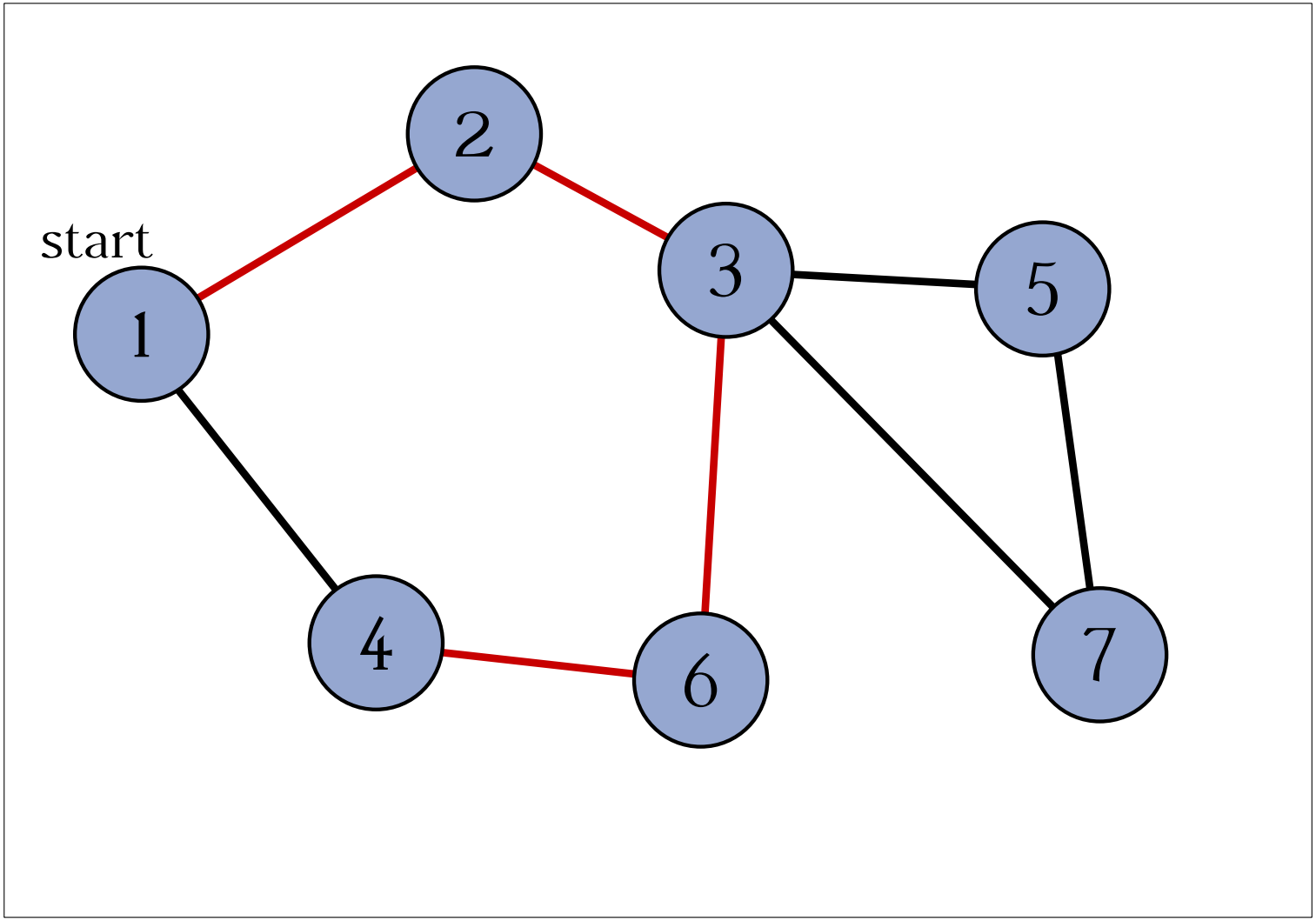
Eulerian cycles

Example:



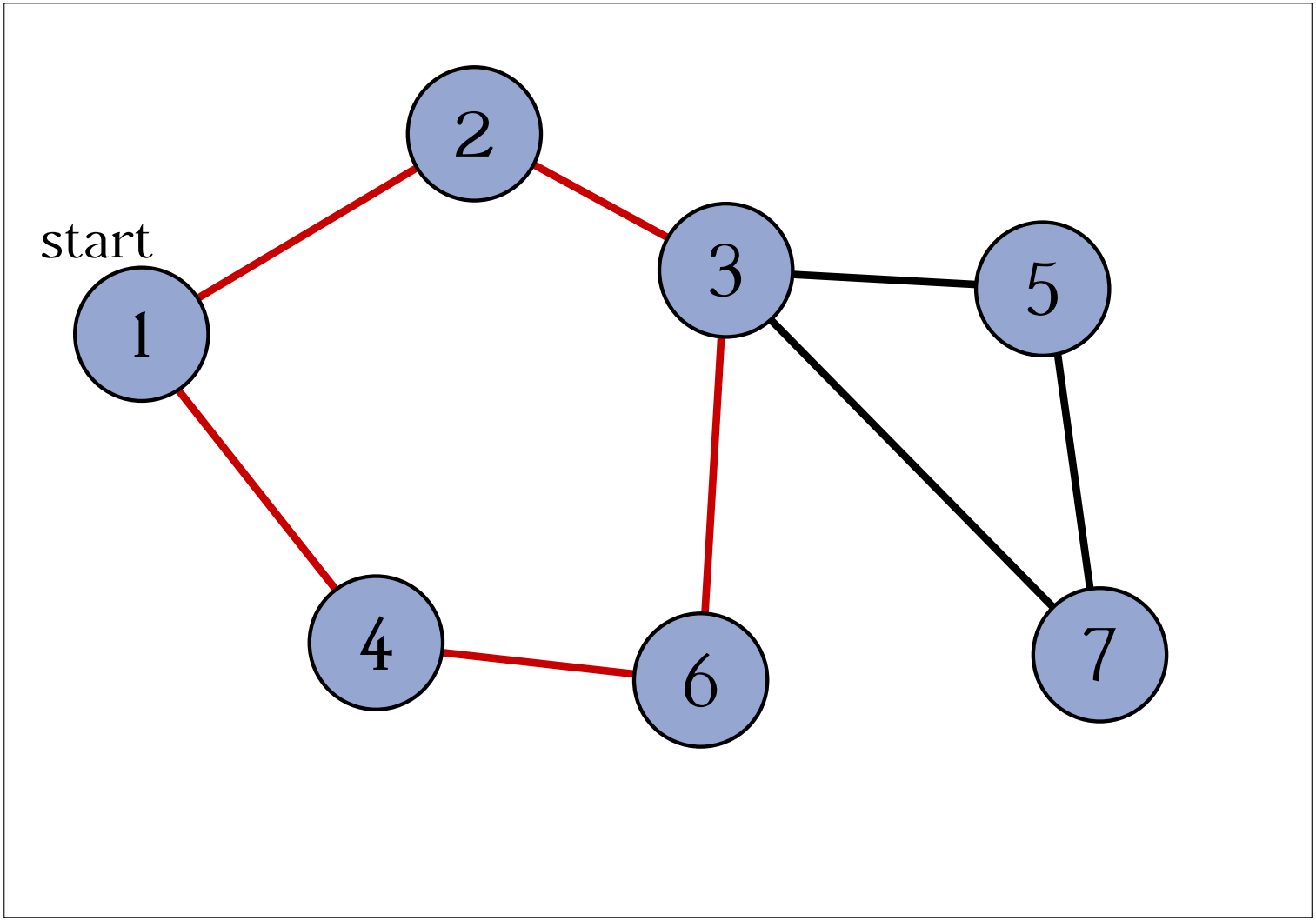
Eulerian cycles

Example:



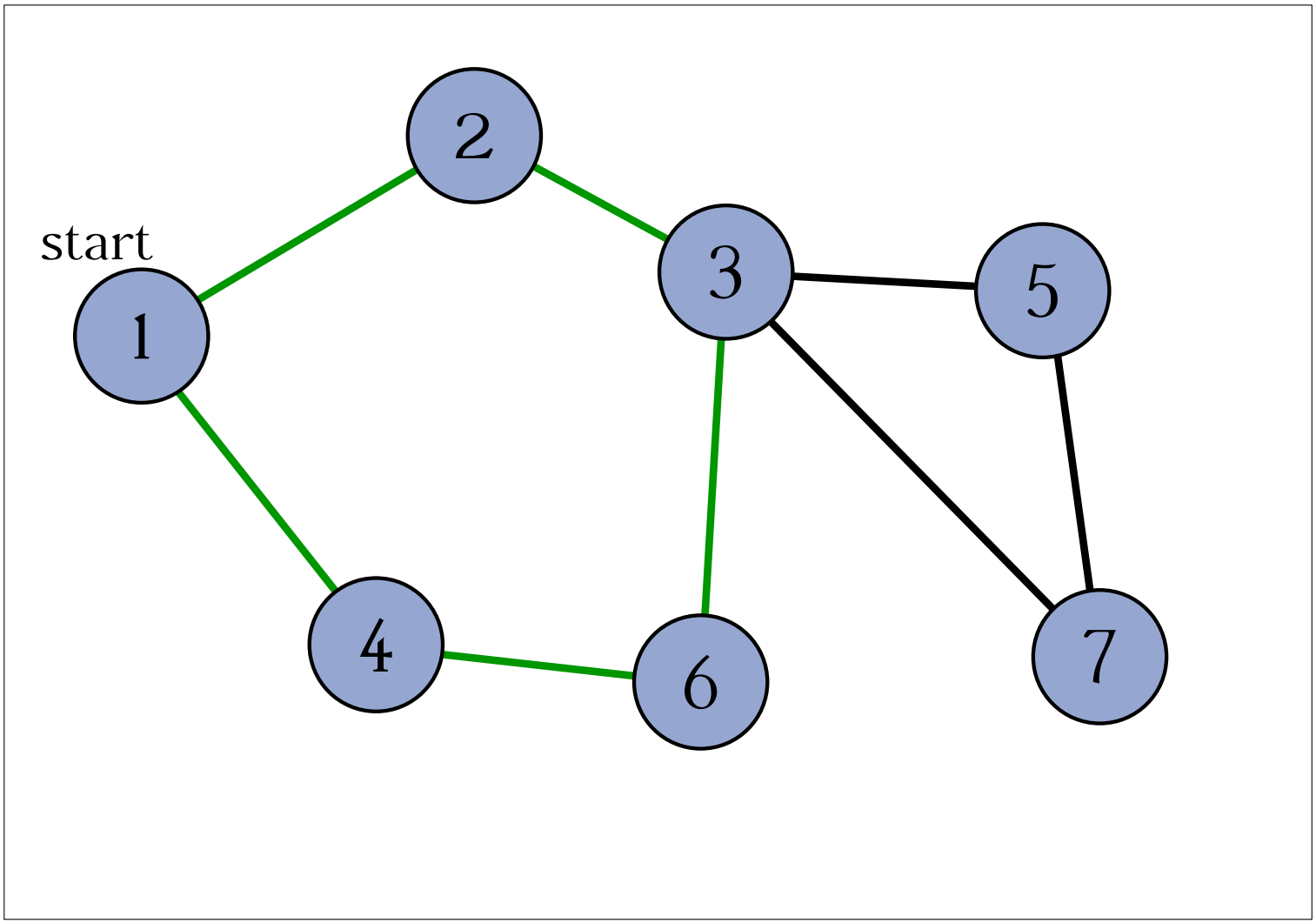
Eulerian cycles

Example:



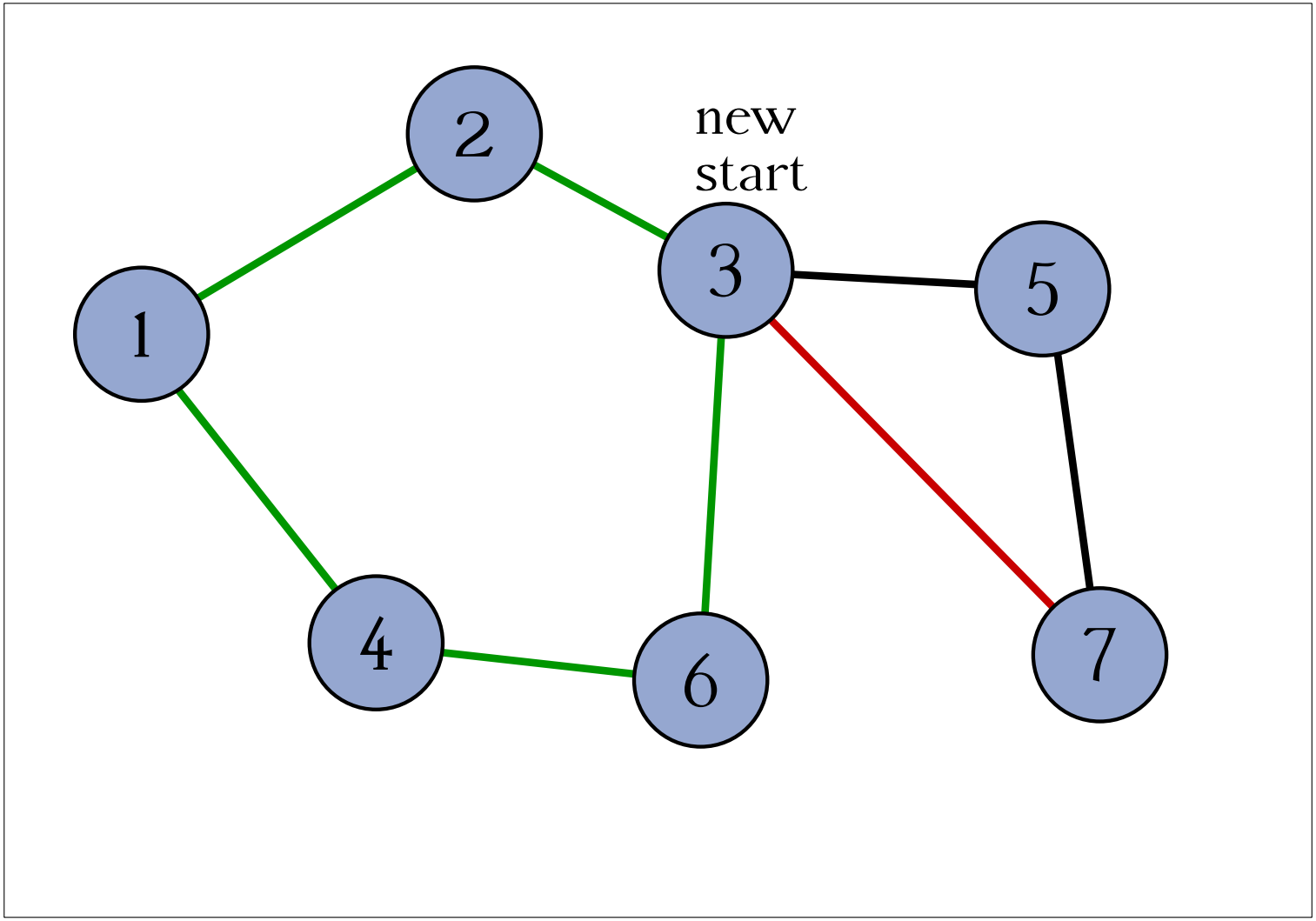
Eulerian cycles

Example:



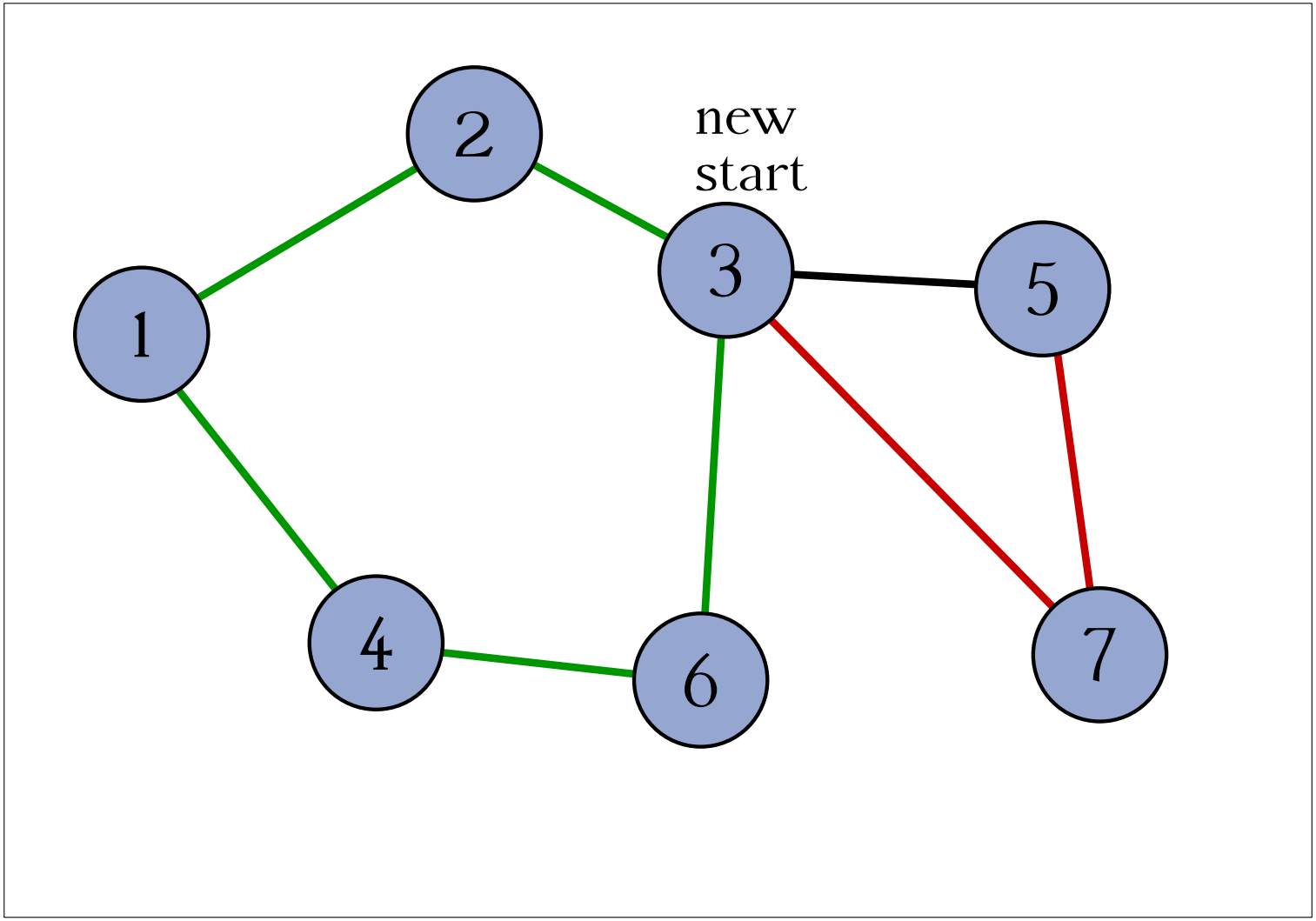
Eulerian cycles

Example:



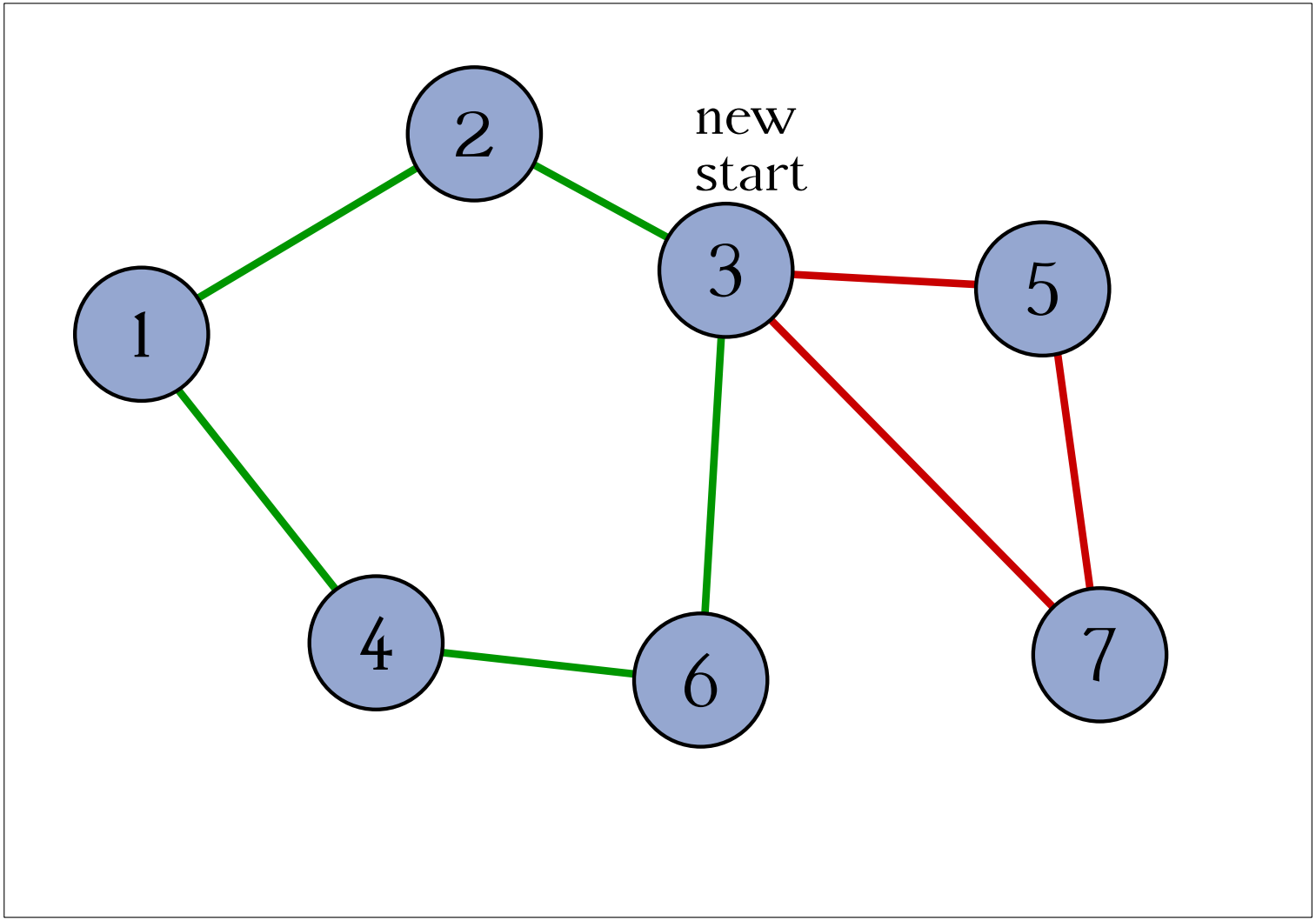
Eulerian cycles

Example:



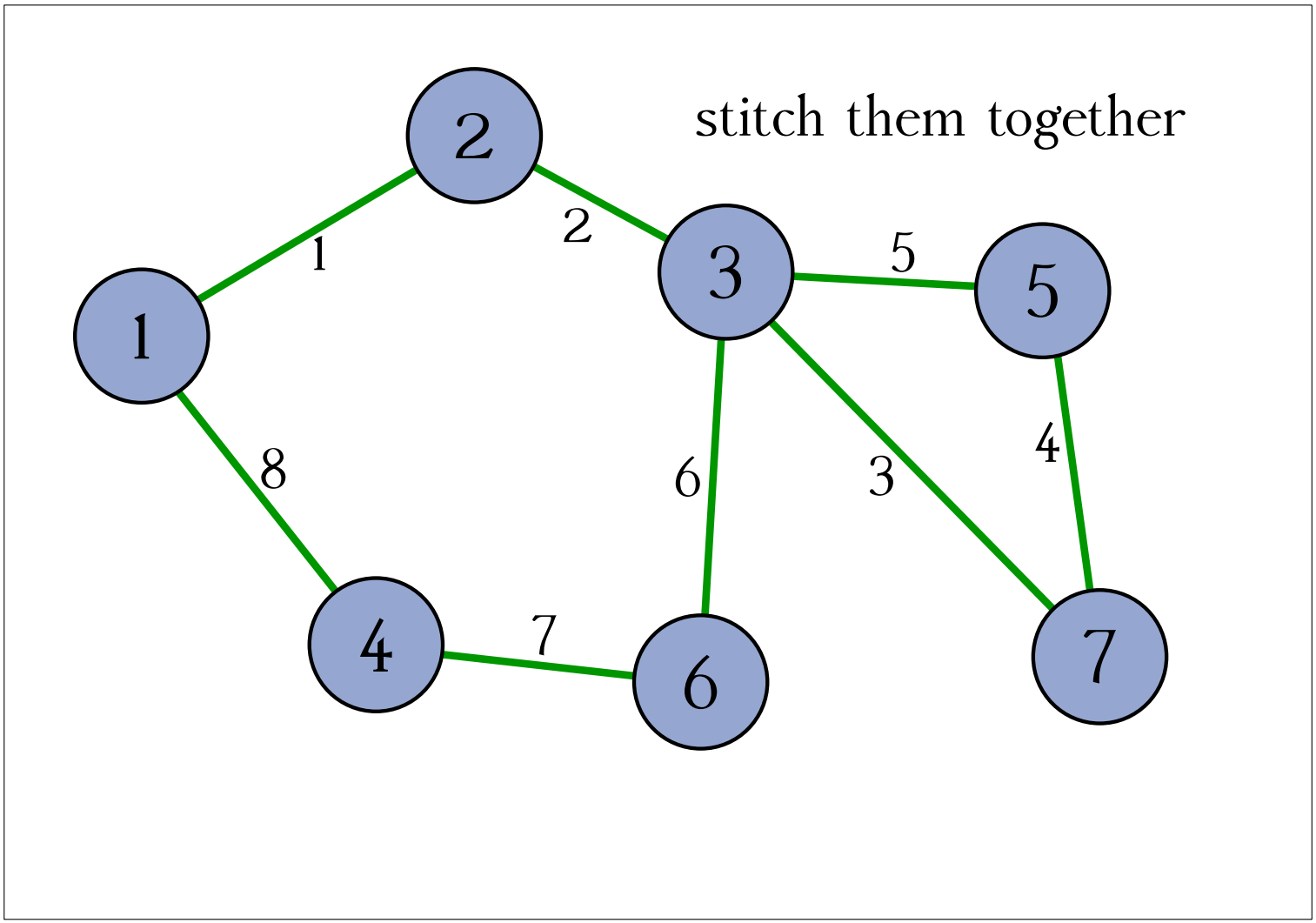
Eulerian cycles

Example:



Eulerian cycles

Example:



Further reading I