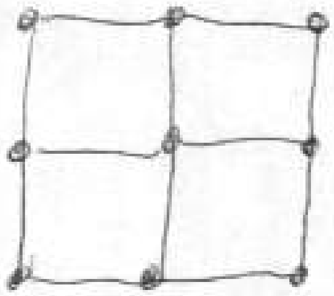
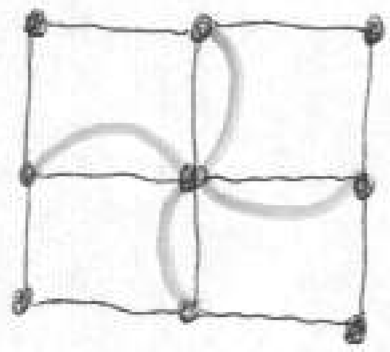
Eulerizing & Semi-Eulerizing Graphs

When a graph modelling a street-routing problem has no Euler circuits or Euler paths, recall that we were still interested in finding an optimal route, i.e. one with the fewest edges overlapped. We show how to solve this.

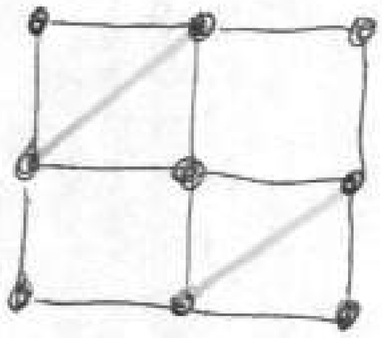
Definition: An Eulerization of a graph is a graph obtained by adding duplicate edges to the original graph in a way such that the result has only even vertices.

Examples: consider the graph

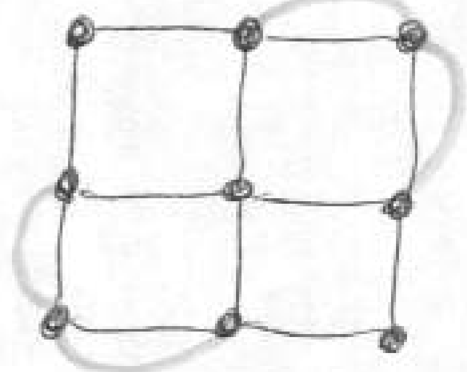




An Eulerization



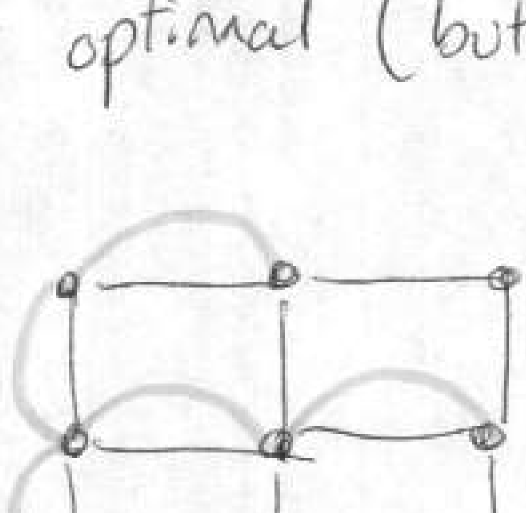
<u>NOT</u> an Eulerization (added edges were not in original graph)



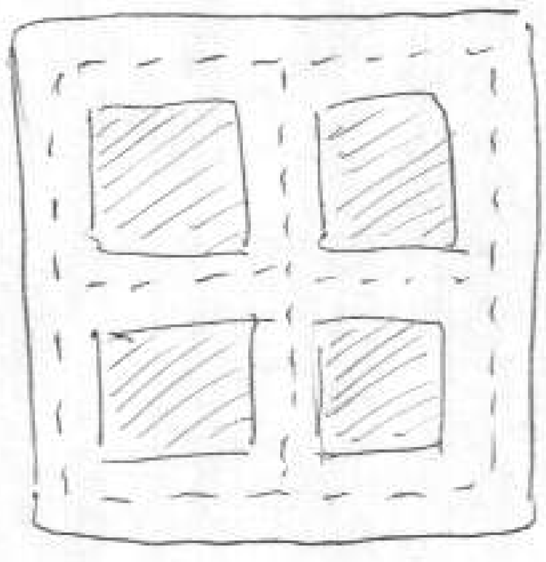
An <u>optimal</u> Eulerization is one which has the minimal possible # of duplicate edges added.

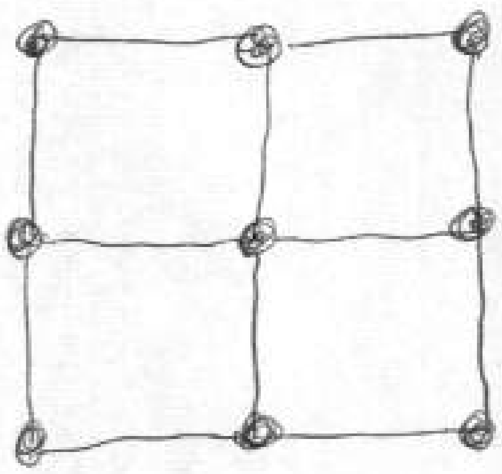
The above two Eulerizations are optimal.

This one is not optimal (but is an Eulerization):



Now imagine we are considering the security guard problem for the very small neighborhood modelled by the above graph: (3)

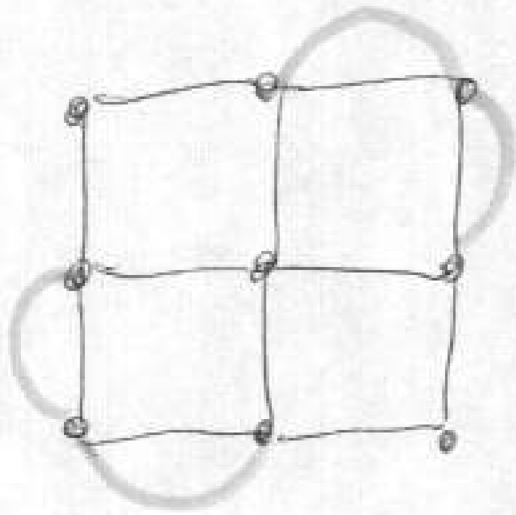




Since the graph has 4 odd vertices, we know that there is no way for the Security guard to choose a route going over each block exactly once (i.e. there are no Euler paths/Euler circuits). The next question is whether we can find a route that is optimal in the sense that it hits every block once and is of the shortest possible length.

For this we Eulerize the graph. The duplicate edges in the Eulerization are the blocks that we decide the security guard will pass over more than once.

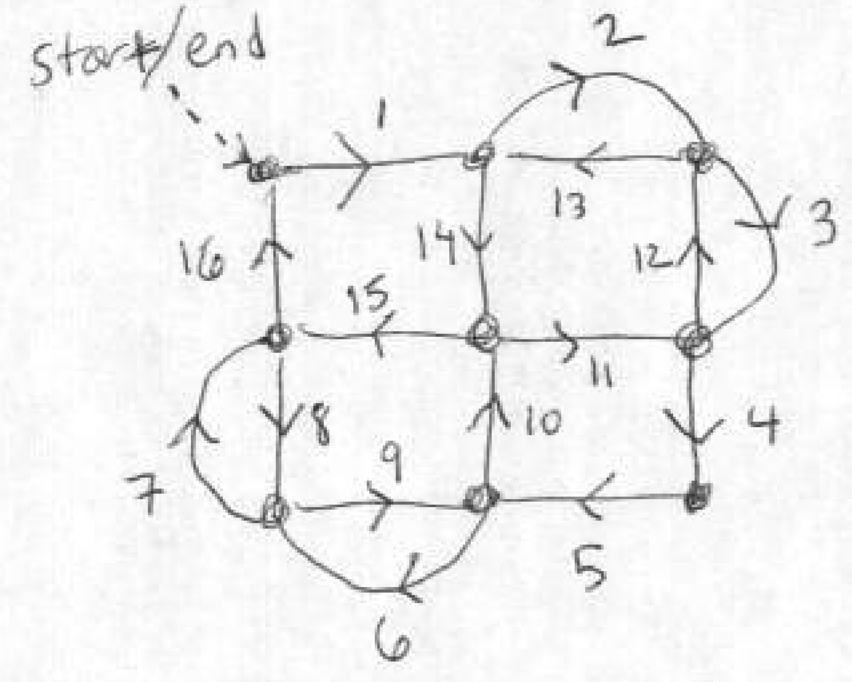
An optimal Eulerization then yields an optimal route in the following way: We saw that the Eulerized graph



is optimal, for example.

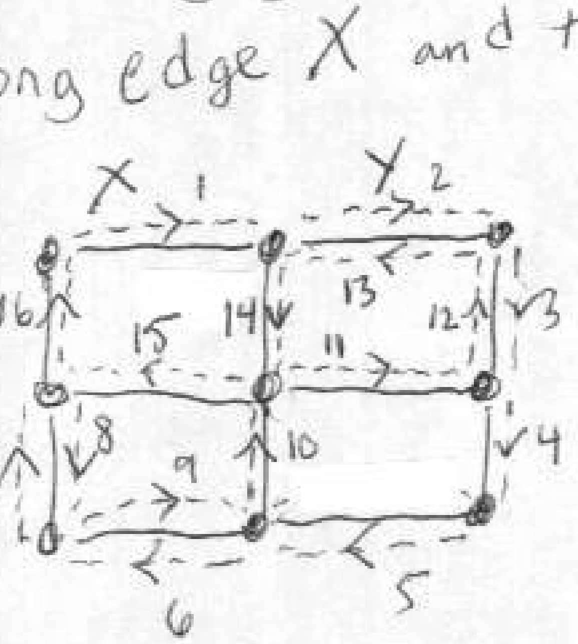
This new graph has only even vertices, so by Euler's Circuit theorem it has an Euler Circuit. We can then use Fleury's Algorithm to find an Euler Circuit.

For example, here's an Euler Circuit:



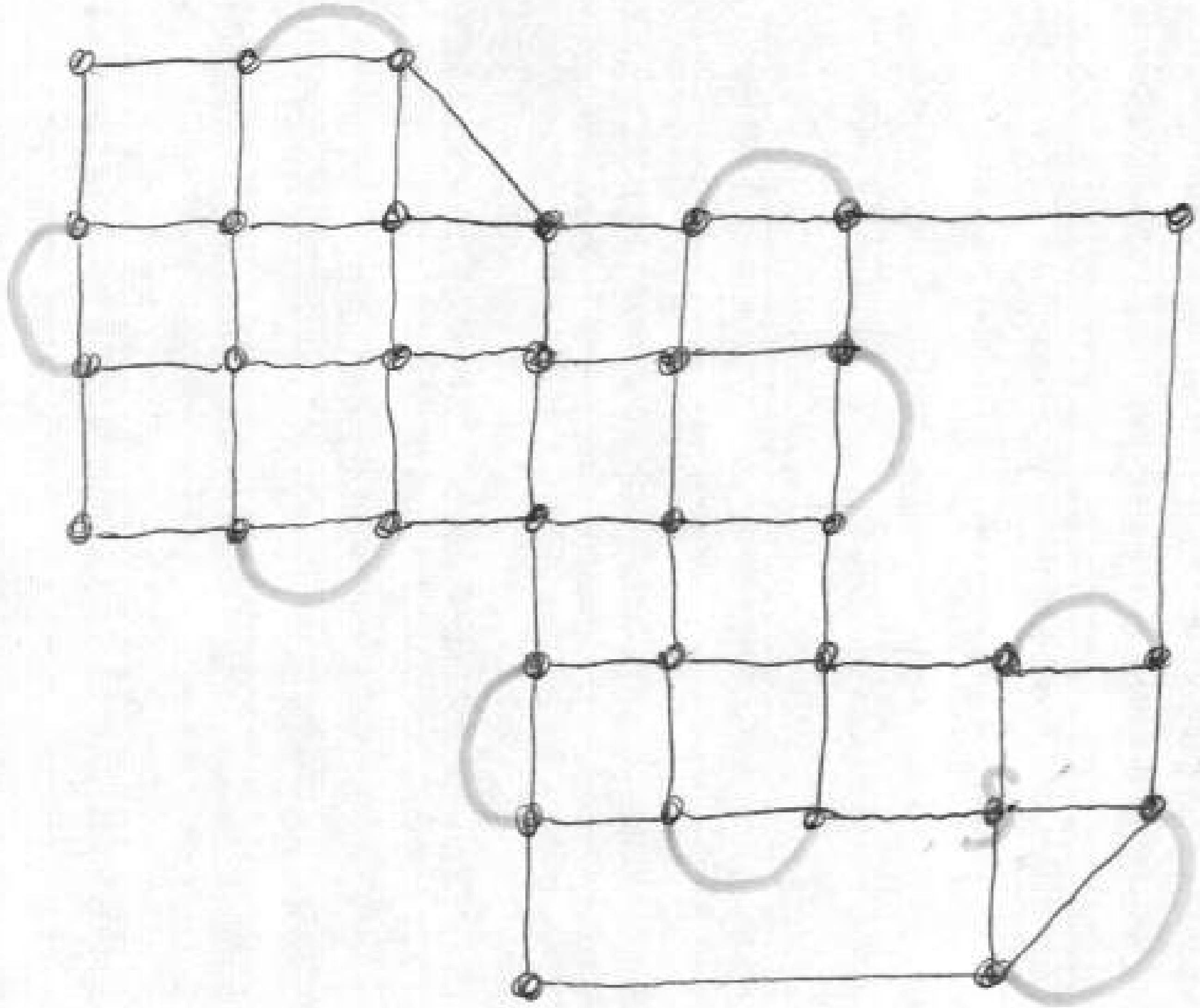
This gives the security guard an optimal route; he/she walks along edge X and then edge Y, but eventually

retraces edge Y in the 13th step of the route.



We can now solve the original security guard problem. 5

"Sunnyside neighborhood" Let's Eulerize it.

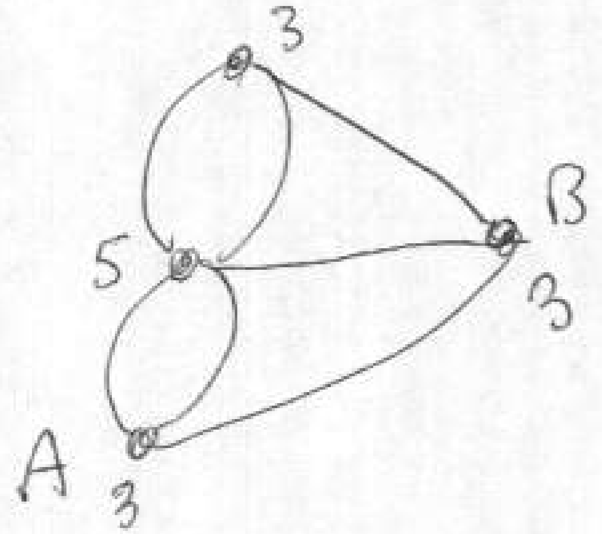


This is optimal since each odd vertex was only touched once by an add duplicate edge, and no even vertices touch duplicate edges.

One then uses Fleury's algorithm to find an Euler Circuit of the Eulerized graph. We'll skip this.

Example Find an optimal route in 1700's Königsberg that crosses all 7 bridges.

Recall that we modelled this problem by the graph

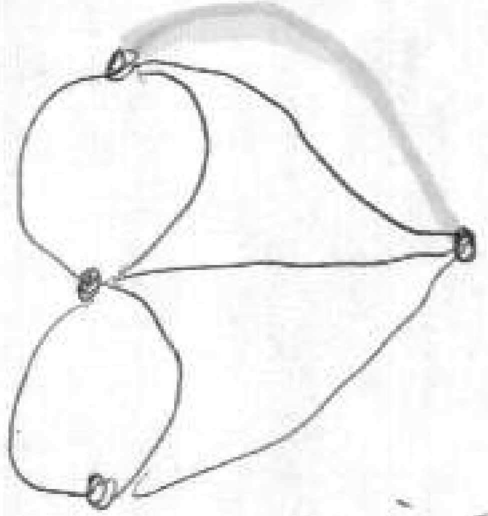


and saw, since it has 4 odd vertices, that it has no Euler Paths or Euler Circuits.

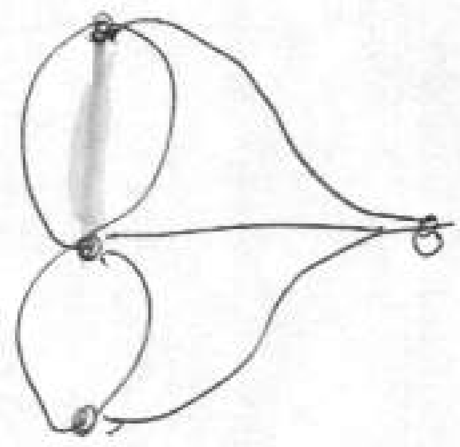
Suppose we are further told that the route must begin at vertex A and end at vertex B.

So this time we are asked to find something different than the previous problem; instead of Eulerizing the graph, we will <u>Semi-Eulerize</u> it. This is much the same as an Eulerization, but the output is not a graph with all even vertices, but rather exactly two odd vertices, so that we can make an Euler path on the new graph.

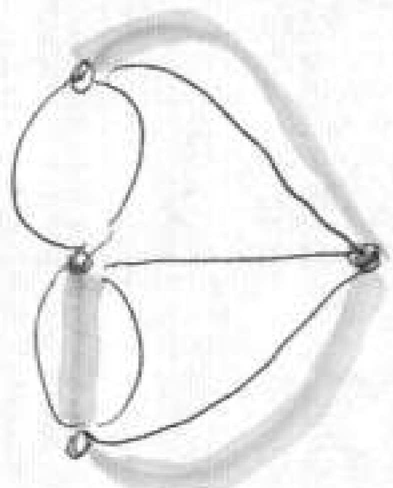
Here are some examples:



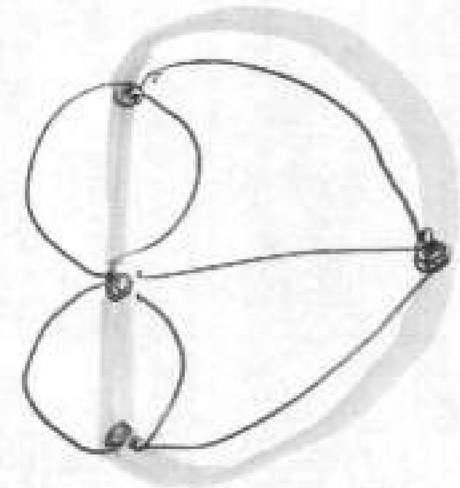
this is a semi-Eulerized graph, but not one for paths from A to B



this one's good for our purpose! Certainly optimal.



this is a semi-Eulerization that works for us, but it is not optimal



not a semi-Eulerized graph; doesn't have two odd vertices

We then use Fleury's Algorithm (or with such a small graph, just quickly look) to find an Euler path: