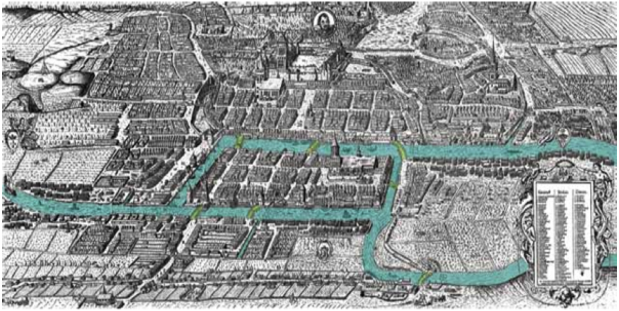
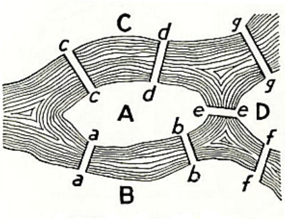
The Königsberg bridge problem



The town of Königsberg, Prussia (now a city in Russia called Kaliningrad) is built on the both banks of the river Preger as well as on an island in the river. At one time there were seven bridges linking one bank to the other as well as both banks to the island. The people in the town wondered if were possible to start at some point in the town, walk about the town crossing each bridge exactly once, and end up back at the starting point.

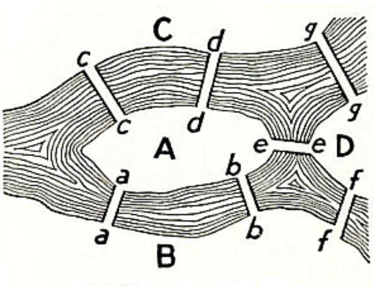
The Königsberg bridge problem

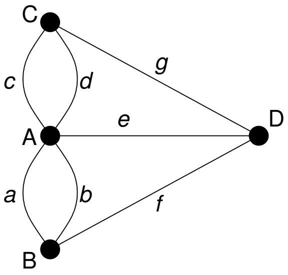


- A, B, C, and D label regions of land (A and D are islands).
- Bridges are labeled with a, b, c, etc.
- Try to find a route, starting anywhere you want, that crosses every bridge once and end up back where you started.

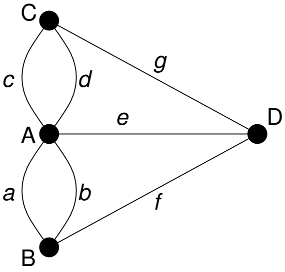
Königsberg bridge graph

- If we focus on the essential parts of this problem, we can construct a graph.
- This *abstraction* is simpler than the bridge picture and yet contains all the necessary information.
- Each region of land is represented by a vertex.
- Each bridge connecting two regions corresponds to an edge.



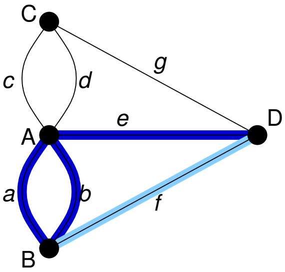


Königsberg bridge graph



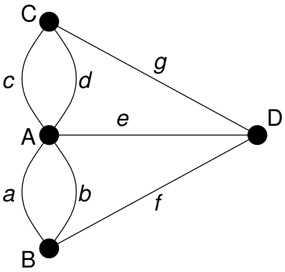
- Consider starting at A then visiting B, D, and C, before returning to A. This does not use each edge, but does visit each vertex.
- We could show this with the walk AaBfDgCcA.

Königsberg bridge graph



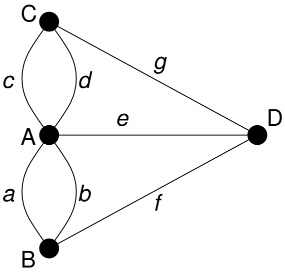
- Suppose we now wanted to use every edge in the graph. Starting at A we can move to B. We can now either return to A or visit D.
- Notice, however, that we will eventually return to B along the third edge connecting to it (since we're trying to use all edges) but then there is no fourth edge to exit from B.

Königsberg bridge graph



- To have any hope of traversing each edge attached to a vertex there must be an even number of edges attached to the vertex; these form an *entry-exit* pair.
- This is also true for the starting vertex, except one pair of edges is an *exit-entry* pair.

Königsberg bridge graph



- Notice what we've established: if there is a vertex with an odd number of edges attached to it, we will be prevented from finding a route that uses all edges once and returns to the starting point.
- If every vertex has an even number of edges attached to it, then there is always an *entry-exit* pair so we can find a route.