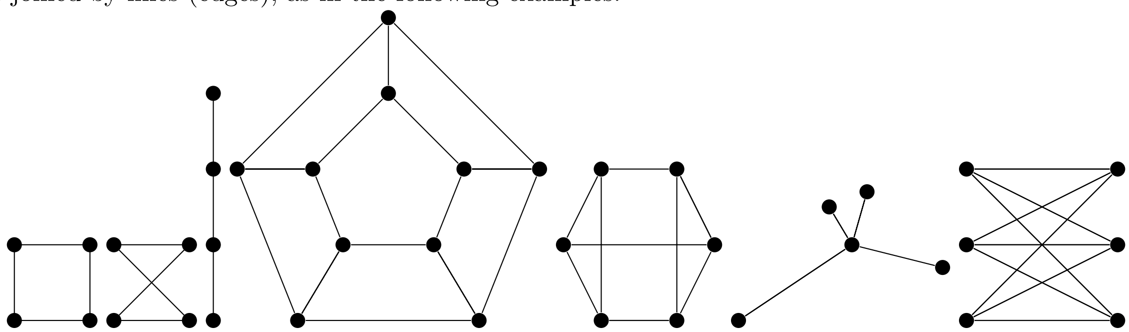
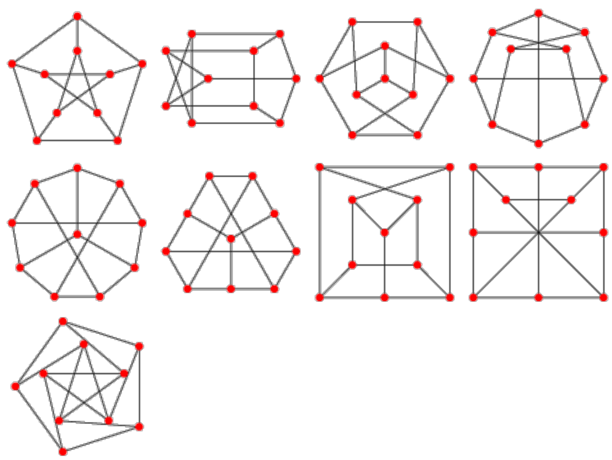
NCEA Level 2 Mathematics
18. Graphs and Networks

The final topic that we will look at is the theory of graphs. A graph is a set of points (called vertices) that are joined by lines (edges), as in the following examples:



As we see at the left, the same graph can be drawn in different ways. The following nine drawings are all of the same graph, the Petersen graph.



In this topic, we will only consider graphs that are *finite*.

Definition.

- Two vertices are *adjacent* if there is an edge joining them.
- The *order* of a vertex is the number of edges incident to it.
- If we label the vertices, then each edge can be identified by its endpoints: if an edge joins a and b, we call the edge ab.
- A *path* on a graph between two vertices a and b is an ordered set of edges $av_1, v_1v_2, \ldots, v_nb$ such that no edge is repeated.
- A *cycle* is a path on the graph between a and itself.
- If every two vertices are connected by some path, then the graph is called *connected*.
- If the graph has no cycles, it is called a *tree*.

Traversibility

Questions about paths and traversibility have been asked about graphs for hundreds of years. In 1736, Leonhard Euler solved the following problem:

The city of Königsberg in Prussia was set on a river, and included two large islands connected by bridges as in the following diagram.

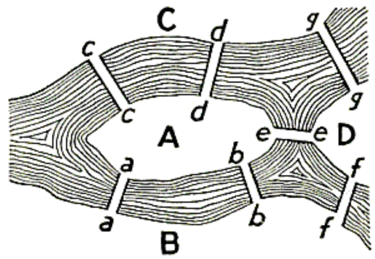
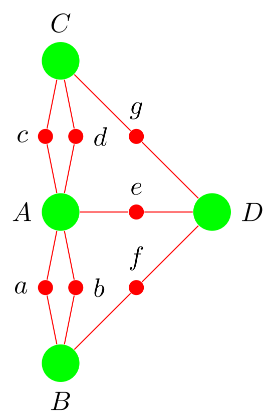


FIGURE 98. *Geographic Map: The Königsberg Bridges.*

Is it possible to walk a path in the city such that every bridge is crossed precisely once?

The answer, as we will see, is no. We begin by drawing a graph of the situation in order to eliminate all of the extraneous details beyond the connectedness of the city.



A path that traverses every edge of a graph exactly once is called an Eulerian path (and an Eulerian path that is also a cycle is called an Eulerian cycle).

Theorem (Euler)**.** *A connected graph has an Eulerian path if and only if it has either zero or two vertices with odd order.*

We will prove the 'only if' half here: that a graph has an Eulerian path only if it has zero or two vertices with odd order. The 'only if' part was proved by Carl Hierholzer in 1873.

Proof of necessity. Suppose a graph has an Eulerian path. Consider some vertex v that the path passes through. Then:

1. If the path does not have an endpoint on v, then v has even order because every edge at the vertex is an edge of the path, and each time the path enters the vertex it leaves (so all the edges at v can be paired up).

2. If the path has precisely one endpoint on v, it has odd order, because all the edges but the endpoint edge can be paired up as in (1).

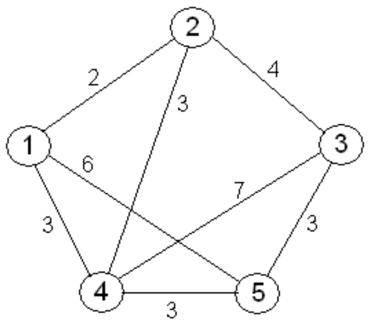
3. If the path has two endpoints at v, then v has even order, because we can pair the two endpoints together and then pair the other edges as in (1).

Every vertex has odd order, except if a vertex falls under case (2). But this can happen at most twice (a path has only two endpoints), and if both endpoints are at different vertices then case (2) applies to both. Hence a graph with an Eulerian path has either zero or two vertices of odd order. $\square$

Hence the Königsberg challenge cannot be solved: there are more than two vertices of odd order.

Weighted graphs

Graphs can be used to model situations in subjects including computer science, scheduling, linguistics, and biology. For example, suppose some company has five distribution centres and wants to find an economical shipping pattern. Let's draw a graph with five vertices and label the edges with the cost of sending a truck (in hundreds of dollars) between the two joined centres:



(If we label edges like this, the graph becomes a *weighted graph.* The numbers are referred to as costs, weights, or distances.)

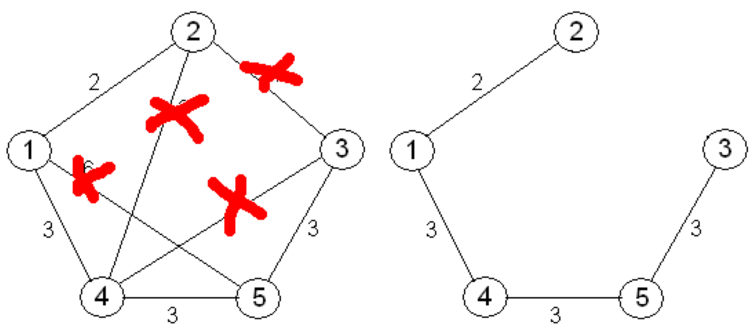
Suppose the company wants to send packages between every pair of distribution centrs, but does not want to pay to run all of the routes graphed. What are the edges that we can safely delete? In practice, we want to find a minimum spanning tree for this graph: a subgraph, without cycles, so that the total cost of the subgraph is a minimum. (Finding this graph would allow this company to minimise expendiature while not losing any connections.) One algorithm to find the minimum spanning tree is the reverse-delete algorithm:

Algorithm (Reverse-delete)**.**

1. List all of the edges in descending weight-order.
2. Examine each edge in order, starting with the most expensive. If deleting this edge disconnects the graph, do not remove the edge; otherwise, remove the edge.

The remaining graph is a minimum spanning tree.

For example, in this case we obtain the following minimum spanning tree:



On the other hand, suppose the company does not run all its routes constantly and instead wants to find the cheapest route for a particular package from one centre to another, utilising any of the eight edges above. This can be done with Dijkstra's algorithm:

Algorithm (Dijkstra)**.** Suppose we want to find the shortest path from some vertex a to some vertex b. In fact, this algorithm will give us the shortest path from a to *any* other vertex!

Label vertex a with zero and every other vertex as ∞; then set a as the current vertex, and:

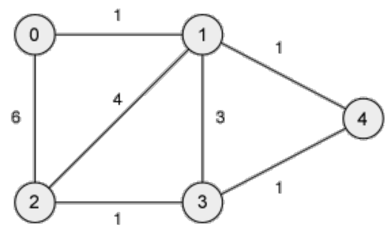
1. Consider every unvisited vertex v directly adjacent to the current vertex, and update the label of v to the smallest value out of:
   - Either the current value on v, or
   - The sum of the current value on the current vertex and the distance from the current vertex to v.
2. If the vertex v has been relabeled by step (1), then mark with an arrow the edge joining the current vertex with v and unmark any other edges on v that have been marked in previous steps.
3. Label the current vertex as visited.
4. If every vertex is visited, then we halt; otherwise, set the current vertex to the vertex with minimal label and repeat from step (1).

Each vertex v is now labeled with the minimum distance from a to v, and the shortest path from a to v is marked by the arrows.

Questions

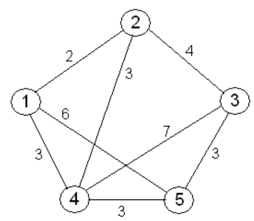
1. Justify, with mathematical reasoning, the following statements.
   (a) A graph that is not connected has no Eulerian paths.
   (b) If a graph has no vertices of odd order, all Eulerian paths are circuits; if there are two vertices with odd order, all Eulerian paths begin at one and end at the other.
   (c) The sum of all the orders of all the vertices must be even.
2. Consider the Königsberg bridge graph.
   (a) Show that it is possible to exhibit a Eulerian path on the graph resulting from adding a single extra bridge.
   (b) Show that, no matter where the bridge is placed, there will still be no Eulerian circuit.
3. Suppose there are three houses on a flat plane (or plain), and each needs to be connected to water, gas, and electricity.
   (a) Show that this is impossible without two connections crossing, without using a third dimension or running a connection through a house.
   (b) Show that, on a torus (a doughnut), it *is* possible.
4. Legend has it that an ancient Indian lord had five sons, and he allowed them to split his land up after his death between them, on the proviso that the land of each son must be in one piece, and must share a boundary with the land of all four other sons. Why is this funny?

5. Consider the following weighted graph G.

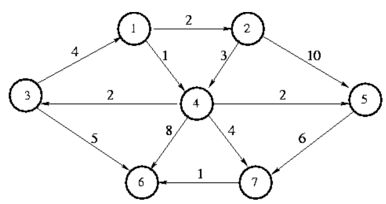


   (a) Give an example of a graph with more than one minimal spanning tree.
   (b) Show that G has precisely one minimal spanning tree.
   (c) Find a vertex v of G such that the minimal spanning tree of the graph also gives the shortest paths from v to every other vertex.

6. Find the cheapest shipping path between distribution centres 1 and 3 in the example of a weighted graph from above (reproduced here).

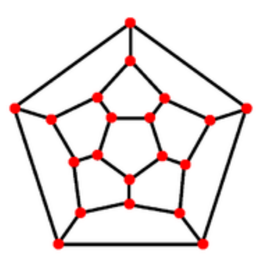


7. Suppose that, in addition to weights on edges of a graph, we assign a direction: perhaps we are modelling the direction and velocity of some fluid flow. Consider the following directed graph:



   (a) Call a directed graph connected if we can give a directed path from every vertex a to every other vertex b (i.e. a path with arrows from a to b). Explain why this directed graph is not connected.
   (b) Find the shortest path from vertex 4 to every other vertex.

8. A **Hamiltonian path** is a path on a graph such that every vertex appears on the path exactly once. If the two endpoints of the path are adjacent, then the path is called a **Hamiltonian cycle**. Find a Hamiltonian cycle on the dodecahedron graph:



   The existence of Hamiltonian paths on graphs is much more difficult than the existence of Eulerian paths. One recent result (from 2005) is that a graph with n vertices has a Hamiltonian path if, for every pair of non-adjacent vertices, the sum of their degrees and the distance between them is greater than n.