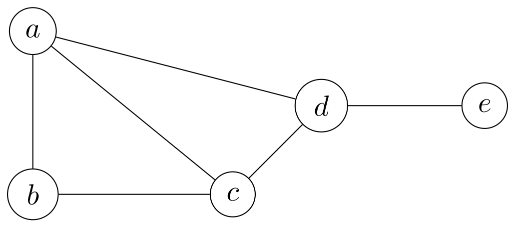

Figure 1: A graph: this is an undirected graph so the edges have no direction.

11- graphs

Graphs are a useful way to represent data; there are powerful algorithms for graphs and so by describing some type of data as a graph it becomes possible to process it using these algorithms. From a more theoretical point-of-view, graphs present interesting problems to the study of algorithms and some branches of mathematics, like combinatorics; many graph theory problems have interesting and elegant solutions while others are unsolved.

A graph is a set of *nodes*, also called *vertices* linked by *edges*. In an undirected graph, these edges have no direction, as in Fig. 1. In a directed graph, the edges each have a direction, as in Fig. 2. In a weighted graph the edges have a weight, as in Fig. 3. Facebook is an undirected graph, since ‘friendship’ is reciprocal, Twitter is a directed graph since ‘following’ is not reciprocal; the distances between cities connected by train lines is a weighted graph. You can imagine other possibilities, like directed weighted graphs and graphs where the nodes also have weights, but we will only consider directed, undirected and weighted graphs here.

A common way to describe a graph algebraically is to use the adjacency matrix. This describes the connections between the nodes; for an undirected graph it is the matrix:

$$A_{ij} = \begin{cases} 1 & i \text{ is connected to } j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

so there is a one in the matrix in the ith row and jth column if the ith node is connected to the jth node. The graphs we have drawn all have letters labeling the nodes, so obviously for the matrix you need to number the nodes, here we’ll just make a node one, b node two and so on, so the adjacency matrix for the graph in Fig. 1 is

$$A = [A_{ij}] = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{2}$$

For a directed graph there is a one in the ith row and jth column of the adjacency matrix if there is a link from i to j, so it is the matrix:

$$A_{ij} = \begin{cases} 1 & i \text{ has an edge pointing to } j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

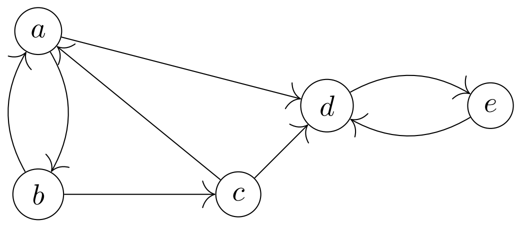

Figure 2: A directed graph.

so, in the case of Fig. 2 the matrix is

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{4}$$

In the weighted graph, as you might expect, the matrix entries correspond to the weights, so

$$A_{ij} = \begin{cases} w & i \text{ is connected with weight } w \text{ to } j \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

so the matrix is

$$A = \begin{pmatrix} 0 & 5 & 6 & 3 & 0 \\ 5 & 0 & 2 & 0 & 0 \\ 6 & 2 & 0 & 8 & 0 \\ 3 & 0 & 8 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{6}$$

For an undirected graph the *degree* of a node is the number of edges incident to it. Thus, for the graph in Fig. 1 the node a has degree three, b has degree two, c three, d three and e has degree one. The degree is an important quantity for describing graphs, though it won't be discussed here, often in applications of graph theory dynamics on a graph is considered, for example, epidemiological models are run on graphs where the nodes are people and the edges contact, so, for a sexually transmitted disease an edge would link two people who had had sex, in a disease that is spread by aerosol it would link two people who had been in the same room. For models like this the average degree of the nodes, and the distribution of degrees, has a strong effect on the dynamics. The degree can also be defined for directed graphs, for a directed graph the degree is the number of incoming edges minus the number of outgoing, so, for the graph in Fig. 2 the degree of a is zero, whereas the degree of b is -1.

Here we will be concerned with paths through graphs, it is possible to give careful definitions of what we mean, but the ideas are intuitive and clear enough to make do with less formal definitions. A *walk* is a series of nodes and edges so that each node is connected to one end of the subsequent edge and the node after that to the other end, that is, it is a path going from node to node along edges without any jumps. A *trail* is a walk in which no edge is repeated. Here a *cycle* is a closed trail, that is a trail that begins and ends at the same point; I say 'here' because the term cycle is sometimes used for a closed walk. Finally, a *tree* is a graph with no cycles; trees are named because they are like trees, but not like the tree in Fig. 4.

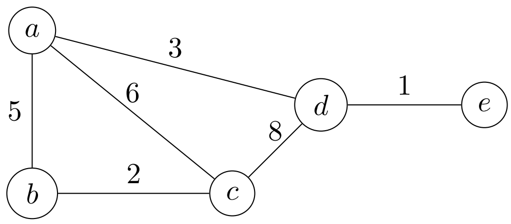

Figure 3: A graph: this is a weighted graph so the edges have a value.

Figure 4: This tree is not a tree, photographed in Hedgemead Park in Bath.

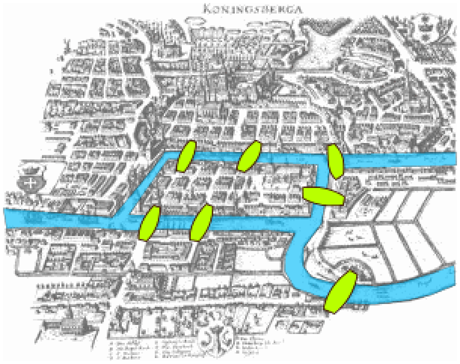

Figure 5: A picture of historical Königsberg with its seven bridges [from wikipedia].

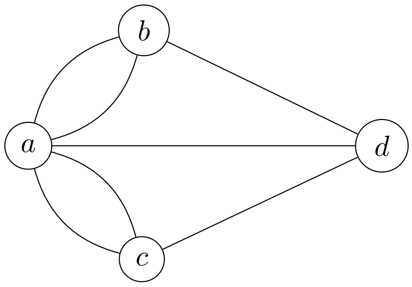

Figure 6: The Bridges of Königsberg graph.

Eulerian trail

The subject of Eulerian trails is usually introduced through Euler's original solution of the problem of the Seven Bridges of Königsberg, Königsberg, now Kalingrad, was a Prussian town where, as the story is told, the townsfolk liked to walk in the evening over the seven bridges in the town that connected a city which spanned two banks of the river Pregel, along with two islands in the river. The layout is shown in Fig. 5. The townsfolk sometimes wondered if there was a route that would allow them to cross all the bridges, but each bridge only once. The great mathematician Euler solved this problem in 1736 by reducing it to a problem in graph theory, he observed that what mattered was the bridges, not the land, so he reduced the four land areas to nodes and thought of the bridges as edges. This graph is shown in Fig. 6.

Now, Euler reasoned like this: the townsfolk might start at one node and end at another, but for the other two nodes they needed to leave as often as they arrived, thus each visit used up two edges, one to arrive by and one to leave. In other words, if you imagine removing edges when the corresponding bridge has been traversed, visiting a node reduced its degree by two, when all the bridges had been crossed the degree of all the nodes must be reduced to zero, so unless a node represents the starting point or end point, it must have an even degree. Thus, if more than two nodes have odd degree, there is no trail that includes every edge. In the case of Königsberg all the nodes are odd, so there is no way the townsfolk can design a path that crosses every bridge exactly once.

These days an *Eulerian trail* is a trail that visits every edge and an *Eulerian cycle* is a cycle

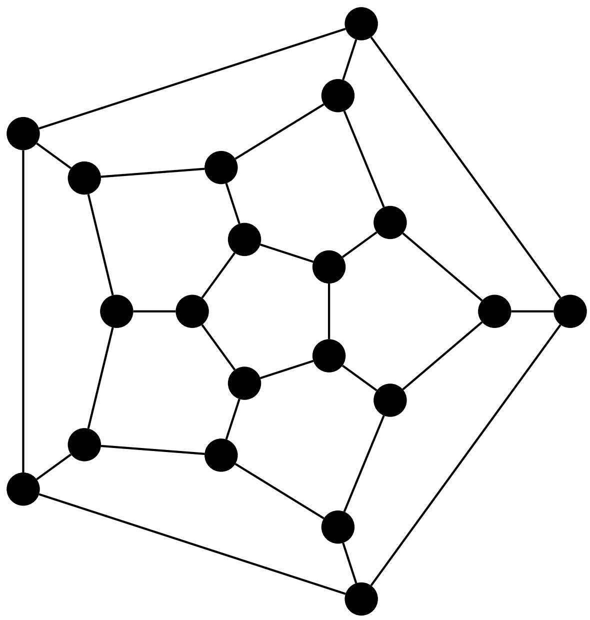

Figure 7: The Hamiltonian Game, a C19 puzzle which challenges you to find a cycle that visits each node exactly once.

that does the same. The discussion of Königsberg above can be reformulated as a Theorem which states that a graph cannot have an Eulerian trail unless all the nodes have even degree or exactly two nodes have odd degree; a graph cannot have an Eulerian cycle unless all the nodes are even. In fact this theorem words the other way too, if a graph has only even degrees there is an Eulerian cycle, if it has two odd degree nodes it has an Eulerian trail starting at one odd node and ending at the other. We won't prove this, or formalize the proof above, but we will consider a construction for the Eulerian trail if there is one. Incidentally Thilo Gross from Eng Math found and walked an Eulerian trail for Bristol, he recounts his walk here: `http://www.reallygross.de/node/81`.

The Eulerian trail uses every edge, this suggests a similar problem, the construction of a trail that visits every node exactly once. This type of trail is called a Hamiltonian trail and the problem of whether a graph has one is hard. Hamilton himself found a Hamiltonian cycle for a graph that has the same connectivity as the surface of dodecahedron: this puzzle is given in Fig. 7.