Graph Theory

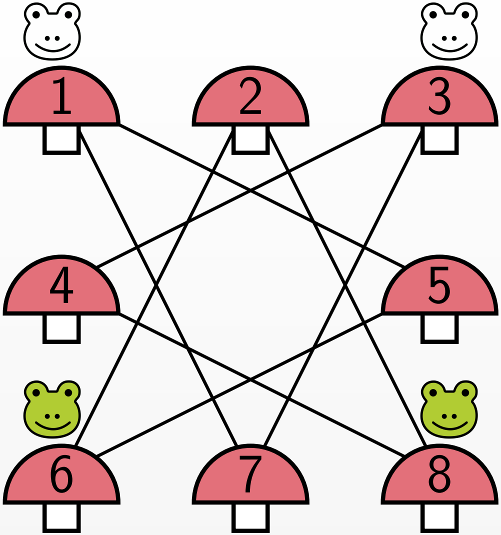
Peter Rowlett

Sheffield Hallam University

p.rowlett@shu.ac.uk

The Four Frogs

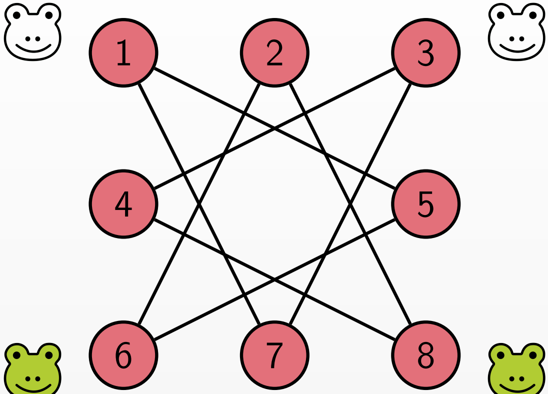
Here are eight toadstools with two green frogs and two white frogs. Jump one frog at a time, in any order, along the the lines between toadstools until the green frogs have switched places with the white frogs. You must not have two frogs on a toadstool at any point. What is the minimum number of moves?



Frog images CC BY-SA 4.0 OpenMoji

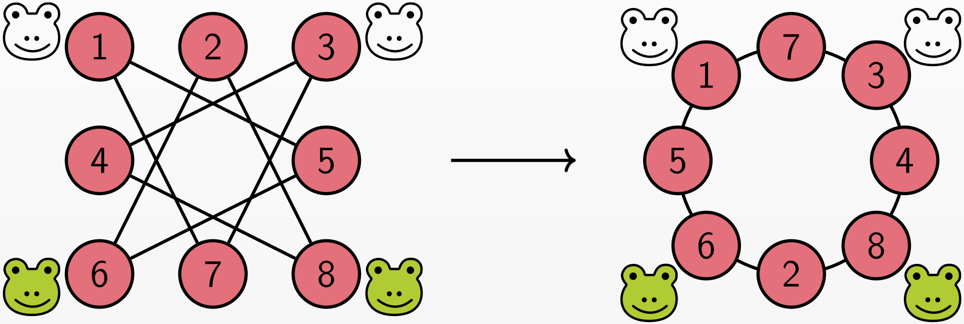
The Four Frogs

- Henry Ernest Dudeney (1857–1930) solved this puzzle in his book *Amusements in Mathematics* (1917).
- He used a method he called his 'buttons and string' method.
- He placed a button on each toadstool, then connected two buttons with string if and only if a frog could move from one to the other.



Buttons and string

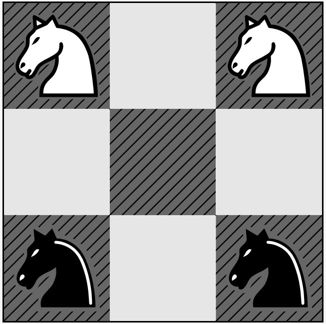
- Dudeney's clever trick is that you can move the buttons around to untangle the strings, then simply jump the frogs round the circle until they have swapped places.



Guarini's Problem

- Actually, this puzzle is one of the earliest involving Chess pieces and was given by Guarini di Forli in 1512.

Two black knights and two white knights are placed at the opposite corners on a 3×3 chessboard. How can the white knights take the place of the black knights, and vice versa, moving according to the rules of Chess?



Buttons and string

- Today, we recognise ‘buttons and string’ as a topological argument.
- In topology we aren’t concerned with the exact placement of objects or their measurements, but with the relationships between objects.
- We would call the ‘buttons and string’ diagram a *graph*, where the buttons are called *vertices* or *nodes* and the strings are called *edges*.
- The graph records which vertices are connected by which edges rather than recording their exact position.
- Then any operation on the graph that does not add or remove vertices or edges between them will still represent the same puzzle.

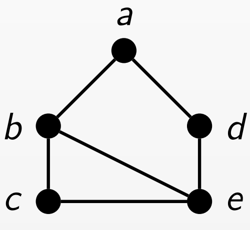
Graphs

- We can think of a graph as being made from a set of vertices, V.
- (Vertices is the plural of vertex.)
- We can then represent an edge between two vertices $a, b \in V$ as $(a, b) \in V^2$, so we make a set of edges $E \subseteq V^2$.
- (Recall $V^2 = V \times V = \{(x, y) \mid x, y \in V\}$.)
- Then we can represent a graph as $G = (V, E)$.

Representing graphs

- We can represent a graph in a diagram by drawing a dot to represent each vertex and drawing a line between two vertices a and b if $(a, b) \in E$, that is if the edge between a and b is in our set of edges.
- For example, let $V = \{ a, b, c, d, e \}$ and $E = \{ (a, b), (a, d), (b, c), (b, e), (c, e), (d, e) \}$.

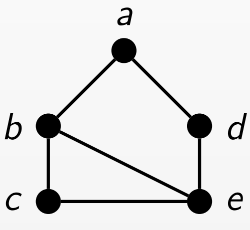
Then we can draw $G = (V, E)$ like this:



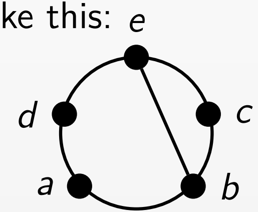
Representing graphs

- We can represent a graph in a diagram by drawing a dot to represent each vertex and drawing a line between two vertices a and b if $(a, b) \in E$, that is if the edge between a and b is in our set of edges.
- For example, let $V = \{ a, b, c, d, e \}$ and $E = \{ (a, b), (a, d), (b, c), (b, e), (c, e), (d, e) \}$.

Then we can draw $G = (V, E)$ like this:

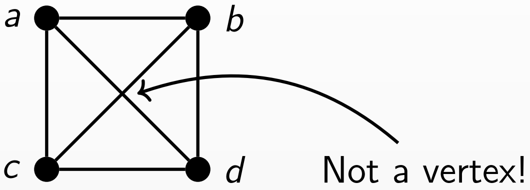


Or, since the exact placement of dots and lines isn't important, any equivalent, like this:



Vertices and crossings

- Note that two edges may appear to cross. These crossings are ignored, they do not form new vertices.



Undirected simple graph

- Let's think about all the edges between pairs of different vertices.
  - Consider two vertices a and b with $a \neq b$.
  - Then we can make a set of all possible edges like this as $K = \{\, (a, b) \mid a, b \in V \text{ and } a \neq b \,\}$.
- We could think about graphs where the set of edges E is equal to this set or a subset of it. Then for a vertex set V we would define our graph as

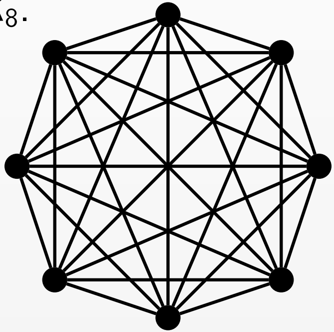
$$G = (V, E); \qquad E \subseteq K.$$

Undirected simple graph

- This type of graph is called an *undirected simple graph*.
- It is undirected because we didn't define the edges as having a direction – we said connecting a to b is the same as connecting b to a.
- And it is simple because it only allows one edge between each pair of vertices and no vertex can connect to itself.

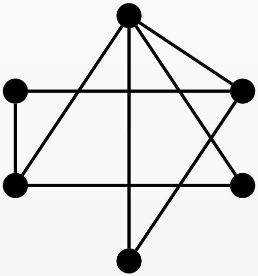
Complete graph

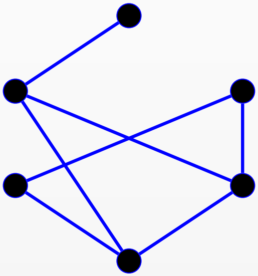
- A simple undirected graph where $E = K$ is called a *complete graph*.
- This means that all possible edges between pairs of distinct vertices are drawn.
- We refer to a complete graph with n vertices as K_n.
- For example, here is K_8.



Complement graphs

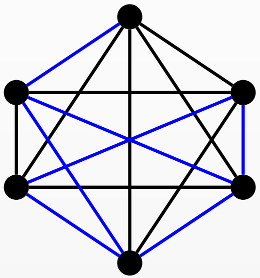
- Say we have a graph G that isn't complete. Then the *complement* graph is a graph on the same vertices so that two vertices are connected by an edge if and only if they aren't connected in G.
- For example, here is a graph and its complement





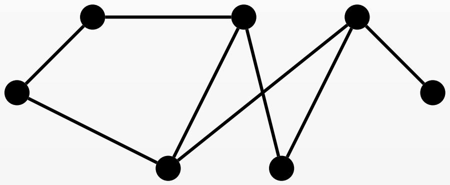
Complement graphs

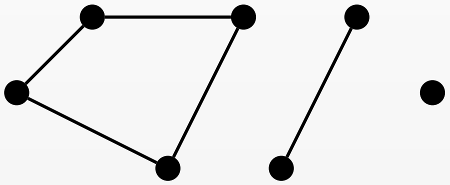
- If we combine the edges of a graph and its complement, we obtain the complete graph with that many nodes.



Connected graphs

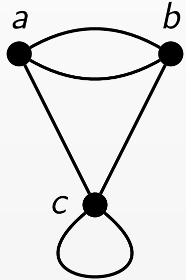
- So far we have seen connected graphs. We can think about graphs that contain parts that are not connected to other parts by edges.
- A *connected graph* is one where no vertices or parts of the graph are isolated from each other.
- For example, the graph on the left is connected and the graph on the right is not.





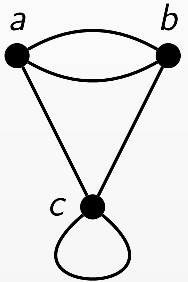
Multiple edges and loops

- So far we have seen simple graphs.
- For non-simple graphs:
  - If two vertices in a graph are joined by two or more edges, these edges are called *multiple edges* (e.g. a and b in the graph below have multiple edges).
  - If an edge joins a vertex to itself, such an edge is called a *loop* (e.g. c in the graph below has a loop).



Vertices

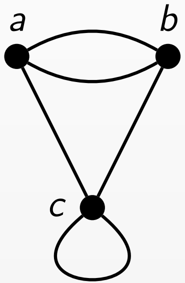
- The *degree* of a vertex is the number of edges incident with it (with a loop counting as two).
- For example,



- a has degree 3.
- b has degree 3.
- c has degree 4.

Odd and even vertices

- If the degree of a vertex is an odd number, we call it an *odd vertex*.
- If the degree of a vertex is an even number, we call it an *even vertex*.
- For example,



- a and b are odd vertices.
- c is an even vertex.