

Planar graphs

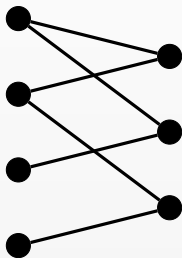
Peter Rowlett

Sheffield Hallam University

`p.rowlett@shu.ac.uk`

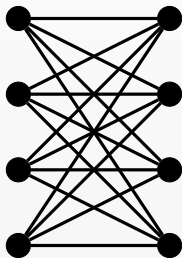
Bipartite graphs

- ▶ A *bipartite graph* is formed from two disjoint sets V_1 and V_2 , where the edges are $E \subseteq V_1 \times V_2$.
- ▶ This means the vertices can be divided into two sets such that the edges only go from a vertex in one set to a vertex in the other.
- ▶ For example,



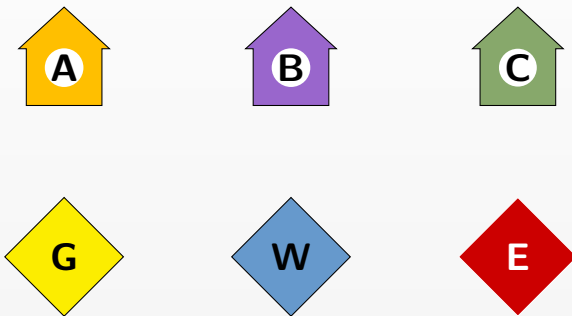
Complete bipartite graph

- ▶ For a bipartite graph with edges connecting vertices in V_1 and V_2 :
 - ▶ if every vertex in V_1 is connected to every vertex in V_2 , we call this a *complete bipartite graph*.
 - ▶ We call this $K_{i,j}$ where $i = |V_1|$ and $j = |V_2|$.
- ▶ For example, here is $K_{4,4}$:



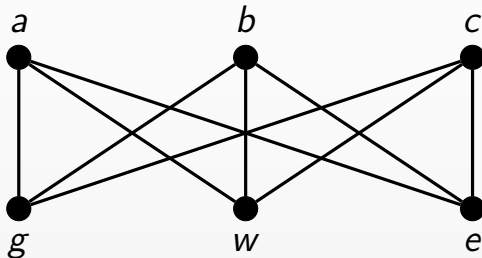
Three Utilities Puzzle

- A classic puzzle: Three houses need to be connected to each of three utilities: gas, water, and electric. The connections have to be direct from house to utility and cannot cross each other. Can this be done?



Three Utilities Puzzle

- ▶ The puzzle asks for a representation of $K_{3,3}$.
- ▶ The puzzle is impossible because $K_{3,3}$ cannot be represented without two edges crossing.

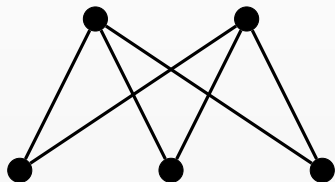


Planar graphs

- ▶ A graph can always be drawn so that no edge crosses itself or crosses an edge joined to one of its vertices, and so that no more than two edges go through any one crossing.
- ▶ The smallest possible number of crossings of a graph drawn this way is its crossing number.
- ▶ A graph with crossing number 0 is called a planar graph.
- ▶ The crossing number for $K_{3,3}$ is 1, hence the impossibility of the Three Utilities Puzzle.

Planar graphs

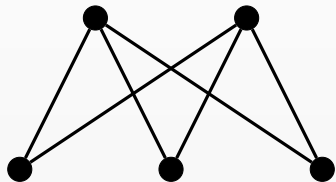
- ▶ The definition of a planar graph is that it can be drawn with no crossings.
- ▶ For example, this is a planar graph:



Planar graphs

- ▶ The definition of a planar graph is that it can be drawn with no crossings.

- ▶ For example, this is a planar graph:



- ▶ To demonstrate that a graph is planar, find a way to draw it without edges crossing.

- ▶ Because it can be redrawn:

