Graph Algorithms

Robin Whitty at www.theoremoftheday.org

Question set for Chapter 9

1. Let $G = (V, E)$ be a graph. A subset $C \subseteq E$ is called an **edge cover** if every vertex has at least one endpoint in C. This is a dual to the idea of a vertex cover whose minimum value is $\tau(G)$; denote by $\varepsilon(G)$ the size of a minimum edge cover for G.

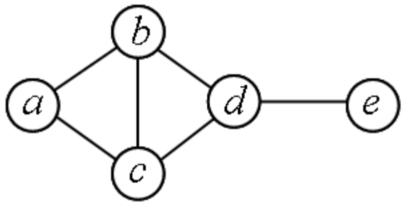
   (a) Find an example of a graph G for which $\varepsilon(G) > \tau(G)$.

   (b) Find an example of a graph G for which $\tau(G) > \varepsilon(G)$.

   (c) Prove that if G is a cycle then $\varepsilon(G) = \tau(G)$.

   (d) Prove that, for any G, we have $\tau(G) \leq 2\varepsilon(G)$.

2. Suppose we want a minimum cover of the graph on the right. The following **linear programme** will achieve this:

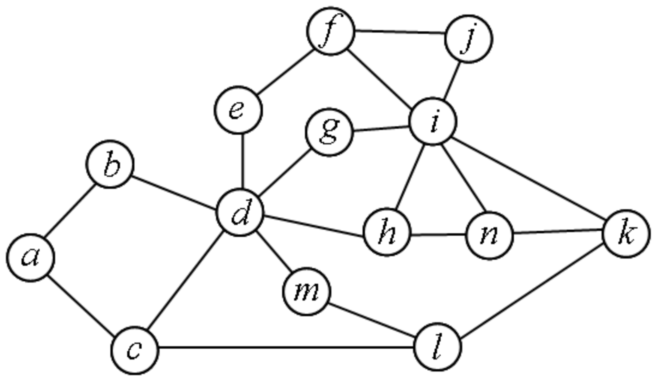


$$\begin{array}{rl} \textbf{minimise:} & a+b+c+d+e \\ \textbf{subject to:} & a + b \geq 1 \\ & a + c \geq 1 \\ & b + c \geq 1 \\ & b + d \geq 1 \\ & c + d \geq 1 \\ & d + e \geq 1 \\ & a, b, c, d, e \geq 0, \text{integer} \end{array}$$

   (a) Find a minimising solution to the inequalities on the left and explain how this corresponds to a minimum cover for the graph.

   (b) What will happen if the requirement that a, b, c, d, e are integer is dropped?

3. Suppose G is a graph and U is a subset of the vertices of G. By the *deletion* of U from G we mean removing from G all the vertices in G together with any edges which are incident with vertices in U. Denote by odd(U) the number of connected components which remain which have an odd number of vertices, when you delete U from G. One of the most celebrated theorems in graph theory[1] says that G has a perfect matching if and only if, for every subset U of vertices, odd(U) $\leq |U|$.
   Use the theorem to prove that the graph above-right has no perfect matching. What if an edge is added joining g to m?



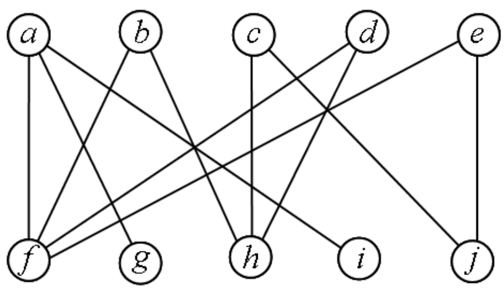
4. Run Kőnig's Bipartite Matching Algorithm for **two** iterations on the graph G below, showing clearly:

   (a) the steps by which a maximal alternating forest F is constructed during each iteration;

   (b) an augmenting path or collection of augmenting paths lying in F;

   (c) the augmented matching in G obtained by using the result of (b).

   Finally perform a **third** iteration and show that it: (1) confirms that a maximum matching has been found; and (2) identifies a minimum cover for G.



[1] Due to Bill Tutte, 1954