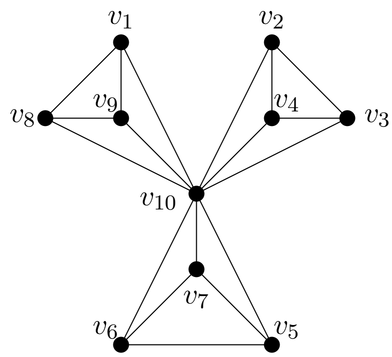
MAS210 Graph Theory Exercises 5

Hand in to BLUE BOX on the GROUND FLOOR of math sci building before 4:30pm on Friday 23/2/07.

Q1 *Consider the following graph G.*



Find a maximum matching in G and a minimum cover of G. Justify the facts that your matching is maximum and your cover is minimum.

Maximum matching: $M_1 = \{v_1v_8, v_2v_3, v_5v_6, v_9v_{10}\}$. (Solution not unique.) Justification. Consider the graph $G - v_{10}$ obtained by deleting the vertex v_{10} and all incident edges from G. Then $G - v_{10}$ has three components each of which is a triangle. Thus, if M is any matching in G, then M can contain at most 3 edges which are not incident to v_{10}. Since M can have at most one edge which is incident to v_{10}, we have $|M| \leq 4 = |M_1|$. Thus M is a maximum matching in G. (Solution not unique.)

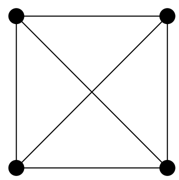
Minimum cover: $U_1 = \{v_1, v_8, v_2, v_3, v_5, v_6, v_{10}\}$. (Solution not unique.) Justification. Let U be any cover of G. We need at least two vertices of U in each triangle of $G - v_{10}$ to cover all the edges of the triangle. We need at least one more vertex in U to cover the edges incident with v_{10}. Thus $|U| \geq 7 = |U_1|$. Thus U_1 is a minimum cover of G. (Solution not unique.)

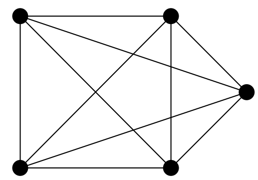
Q2 *The* complete graph K_n *is the graph with n vertices in which each vertex is joined to every other vertex by an edge. The* complete bipartite graph $K_{m,n}$ *is the graph with vertices partitioned into two sets X, Y where $|X| = m$, $|Y| = n$, and in which each vertex of X is joined to every vertex of Y by an edge.*

(a) Draw K_4, K_5, $K_{3,3}$, and $K_{3,4}$.
(b) Determine match(K_n), cov(K_n), match($K_{m,n}$), and cov($K_{m,n}$) for all integers $1 \leq m \leq n$. Justify your answers.

(a)

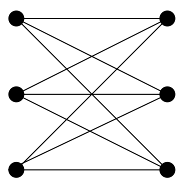
K_4

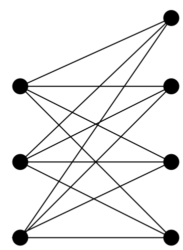




K_5

$K_{3,3}$





$K_{3,3}$

(b) $match(K_n) = \lfloor n/2 \rfloor$.
Justification. Let $V(K_n) = \{v_1, v_2, \ldots, v_n\}$. Let M be a maximum matching in K_n. Since the edges in M have no common end vertices we have $2|M| \leq n$. Thus $match(K_n) = |M| \leq \lfloor n/2 \rfloor$. If n is even, say $n = 2t$, then $M_1 = \{v_1v_2, v_3v_4, \ldots, v_{2t-1}v_{2t}\}$ is a matching in G of size $\lfloor n/2 \rfloor = t$. If n is odd, say $n = 2t+1$, then $M_1 = \{v_1v_2, v_3v_4, \ldots, v_{2t-1}v_{2t}\}$ is a matching in G of size $\lfloor n/2 \rfloor = t$. Thus $match(K_n) = \lfloor n/2 \rfloor$.

$cov(K_n) = n - 1$.
Justification. Let U be a minimum cover of K_n. If $|U| \leq n-2$ then we may choose $v_i, v_j \in V(K_n) - U$, and U will not cover the edge v_iv_j. Thus $covK_n = |U| \geq n - 1$. Since $U_1 = \{v_1, v_2, \ldots, v_{n-1}\}$ is a cover of G of size $n-1$, we have $cov(K_n) = n - 1$.

$match(K_{m,n}) = m = cov(K_{m,n})$.
Justification. Let $V(K_{m,n}) = X \cup Y$ where $X = \{x_1, x_2, \ldots, x_m\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$. Then $M = \{x_1y_1, x_2y_2, \ldots, x_my_m\}$ is a matching in $K_{m,n}$ and $X = \{x_1, x_2, \ldots, x_m\}$ is a cover of $K_{m,n}$. Since $|M| = m = |X|$, Corollary 5.1.7 from the notes implies that M is a maximum matching of $K_{m,n}$ and X is a minimum cover of $K_{m,n}$.

Q3 *(a) Write down the iterative step in Dijkstra's algorithm.*
(b) Prove that the time taken by the $(i+1)$'th iteration of Dijkstra's algorithm applied to a network N is $O(|V(N)| + d_N(x_{i+1}))$, assuming that all elementary arithmetic operations can be performed in constant time no matter how large the numbers involved are.
(c) Deduce that, under the same assumption, the total time taken when Dijkstra's algorithm is run on N is $O(|V(N)|^2 + |(E(N)|)$. Hence deduce that Dijkstra's algorithm is strongly polynomial.

(a) [Iterative Step] Suppose we have constructed a tree T_i with $V(T_i) = \{x_1, x_2, \ldots, x_i\}$ for some $i \geq 1$ and have labelled all vertices $y \in V(N) - V(T_i)$ with a label $label_i(y) = [x, h_i(y)]$.

- If $V(T_i) \neq V(N)$ then choose a vertex $\hat{y} \in V(N) - V(T_i)$ such that $label_i(\hat{y}) = [\hat{x}, h_i(\hat{y})]$ and $h_i(\hat{y})$ is as small as possible. Put $x_{i+1} = \hat{y}$ and $T_{i+1} := T_i + x_{i+1} + \hat{x}x_{i+1}$. For each vertex $y \in V(N) - V(T_{i+1})$ put

$$label_{i+1}(y) = \begin{cases} [x_{i+1}, h_i(x_{i+1}) + w(x_{i+1}y)] & \text{if } y \text{ is adjacent to } x_{i+1} \text{ and} \\ & h_i(x_{i+1}) + w(x_{i+1}y) < h_i(y), \\ label_i(y) & \text{otherwise.} \end{cases}$$

- If $V(T_i) = V(N)$ then STOP. Put $T = T_i$ and output T.

(b) In the $(i+1)$'th iteration we first grow the tree T_i. This involves looking at the labels on all vertices in $V(N) - V(T_i)$ and choosing the vertex $\hat{y}$ such that $h_i(\hat{y})$ is as small as possible. Since $|V(N) - V(T_i)| < |V(N)|$, and since elementary arithmetic operations take constant time, the time taken will be $O(|V(N)|)$. We then update the labels. This involves looking at all edges incident to x_{i+1}, calculating the sum of two numbers and comparing a previous vertex label to this sum, so the time taken will be $O(d_N(x_{i+1})$. Hence the total time taken in the $(i+1)$'th iteration is $O(|V(N)|+d_N(x_{i+1})$.

(c) Since the number of iterations is $|V(N)|$, the total time taken by the algorithm is

$$O\left(|V(N)|^2 + \sum_{i=0}^{|V(N)|-1} d_N(x_{i+1})\right) = O\left(|V(N)|^2 + |E(N)|\right),$$

since $\sum_{i=0}^{|V(N)|-1} d_N(x_{i+1}) = 2|E(N)|$.

Since $|V(N)|^2 + |E(N)|$ is a polynomial in $|V(N)|$ and $|E(N)|$, the algorithm is strongly polynomial.