UNIVERSITY OF TORONTO – CSC 373 – JULY 26, 2007
TEST 2

Student Number: ______________________

Email Address: ______________________

Last (Family) Name(s): ______________________

First (Given) Name(s): ______________________

- This test is worth 15% of your final mark.
- Answer each question directly on the test paper, in the space provided. Use the reverse side of the pages for rough work. If you need more space for one of your solutions, use the reverse side of a page and *indicate clearly the part of your work that should be marked*.
- **20% rule:** If you are unable to answer (a part of) a question, you will get 20% of the marks for the (part of the) question if you write "I don't know" and *nothing else* for that part/question.

Question 1 (out of 13)	
Question 2 (out of 11)	
Total (out of 24)	

Question 1 [13]

Recall that a *bipartite graph* $G = (V, E)$ is an undirected graph whose set of vertices V can be partitioned as $V = X \cup Y$ (where $X \cap Y = \varnothing$) with the property that every edge $e \in E$ has one end in X and the other end in Y. A *matching* M in G is a subset of the edges $M \subseteq E$ such that each vertex appears in at most one edge in M. The Bipartite Matching Problem is that of finding a matching in G of largest possible size. (In the lecture we considered the case where the partitions (X, Y) of V have equal size. The case we consider here is more general, but the problem is NOT harder.) You are asked to solve this problem using **flow network**.

a [4] Given a bipartite graph G, construct a flow network that you will use to solve the Bipartite Matching Problem for G. Clearly specify the vertices, edges and capacities of the network.

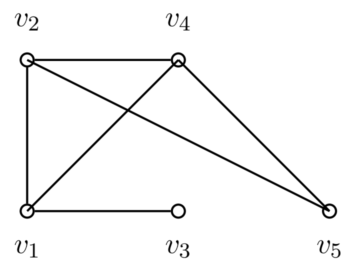
b [2] Give the name of an algorithm you learned in lecture that returns a maximum flow for the network you gave in part **a**. What is the running time of that algorithm? Give your answer using $\mathcal{O}$ notation.

c [4] Show how to obtain a matching of maximum size for G using the maximum flow returned by the algorithm you gave in part **b**.

d [3] Briefly argue that the matching you obtained in part **c** has maximum size.

Question 1 continues here

Question 2 [11] A *vertex cover* of an undirected graph $G = (V, E)$ is a subset V' of V ($V' \subseteq V$) such that for each edge $(u, v) \in E$ at least one of u and v belongs to V'. For example, in the figure below $\{v_1, v_2, v_4\}$ is a vertex cover. Another vertex cover is $\{v_2, v_3, v_4\}$. But $\{v_1, v_2, v_3\}$ is NOT a vertex cover, because it does not contain any endpoint of the edge (v_4, v_5).



Given an undirected graph $G = (V, E)$ and an integer k, the Vertex Cover Problem is to decide whether there is a vertex cover V' of G that has at most k vertices ($|V'| \leq k$). (The Vertex Cover Problem is a special instance of the Set Cover Problem discussed in lecture: each edge can be viewed as the 2-elements subset of its endpoints.) In this question you are asked to formulate the Vertex Cover Problem as a 0-1 IP. (You are NOT asked to solve the 0-1 IP.)

a [2] Given an undirected graph G and an integer k, specify the variables for the 0-1 IP and explain their meaning.

b [2] Specify the objective function. Clearly indicate whether it is to be maximized or minimized.

c [4] Specify the constraints.

d [3] How do you use the optimal value of the objective function to solve the Vertex Cover Problem for input G and k? Briefly justify.