Name: _____

Entry number: _____

There are 5 questions for a total of 75 points.

---

1. Solve the following two problems:
   (a) (5 points) An edge in a network flow graph is called *downwards critical* if decreasing the capacity of this edge decreases the maximum flow in the network. Design an algorithm to find a downwards critical edge in a network. Give pseudocode, discuss running time and give proof of correctness.

(b) (5 points) An edge in a network flow graph is called *upwards critical* if increasing the capacity of this edge increases the maximum flow in the network. Design an algorithm to find an upwards critical edge in a network in case there exists one. Give pseudocode, discuss running time and give proof of correctness.

2. (10 points) You are given n pairs of integers $(d_1, d'_1), ..., (d_n, d'_n)$ such that $\forall i, d_i, d'_i \geq 0$. You have to design an algorithm that determines if there exists a directed graph $G = (\{1, ..., n\}, E)$ such that the in-degree of vertex i is d_i and the out-degree of vertex i is d'_i. Your algorithm should also output a graph with the given degree sequence, in case there exists one. Give pseudocode, discuss running time and proof of correctness.

3. (15 points) There is an $n \times n$ grid in which some cells are empty and some are filled. The empty/filled cells are given by an $n \times n$, 0/1 matrix F. Cell (i, j) is empty iff $F[i, j] = 0$. You have unbounded supply of 2×1 tiles (called dominoes). Each domino could be placed on the empty cells of the grid in horizontal and vertical manner (note that you need two consecutive empty cells on the grid for doing this). You have to design an algorithm to determine if the grid can be covered by placing these 2×1 dominos such that each empty cell is covered by exactly one domino. Give pseudocode, discuss running time and give proof of correctness.

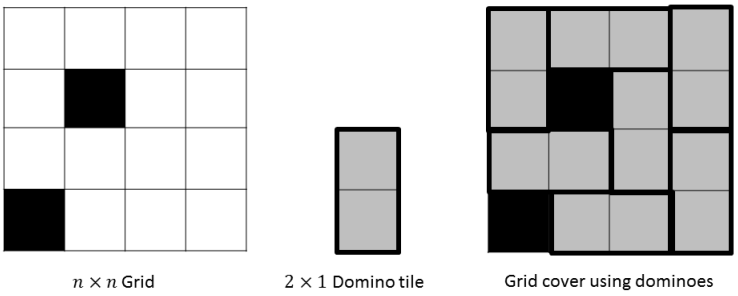

Figure 1: Example of a 4×4 grid that can be covered using 2×1 domino tiles.

4. (20 points) Given an s-t flow network G, design an algorithm to determine if there is a *unique* s-t min-cut in G. Give pseudocode, discuss running time, and give proof of correctness.

5. (20 points) Given a bipartite graph $G = (L, R, E)$ with $|L| = |R| = n$, an edge e is said to be *imperfect* iff there does not exist any perfect matching in G that includes e. Design an algorithm to output all imperfect edges of the given graph G. Give pseudocode, discuss running time, and give proof of correctness.