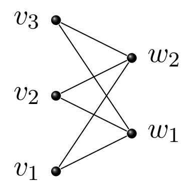
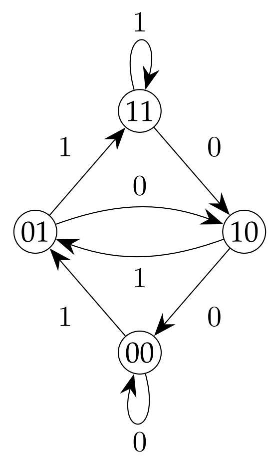
MATH 113: DISCRETE STRUCTURES
HOMEWORK DUE WEDNESDAY WEEK 8

Problem 1. Let m and n be positive integers. Define the graph $C(m,n)$ with $m+n$ vertices: $v_1, \ldots, v_m$ and $w_1, \ldots, w_n$. There is an edge in $C(m,n)$ connecting each v_i to each w_j. For example, $C(3,2)$ is pictured below:



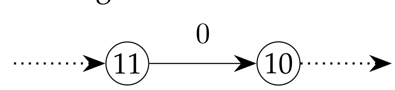
(a) For which values of m and n does $C(m,n)$ have a closed Eulerian walk? Explain.
(b) For which values of m and n does $C(m,n)$ have a Hamiltonian cycle. Explain. (Once you find an argument to restrict m and n to certain values, you still need to demonstrate that for those values, a Hamiltonian cycle exists.)

Problem 2. Let G_2 be the directed graph with edges labeled by 0 and 1 pictured below:



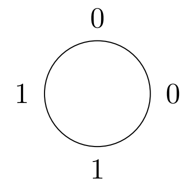
We say G_2 is *directed* because each edge has been assigned a direction. A path or walk in a directed graph is the same as before, however, you can only travel along an edge in its direction. The *outdegree* of a vertex v is the number of edges leaving v, and the *indegree* of v is the number of edges entering v. The theorem for the existence of closed Eulerian walks from our text (for undirected graphs) generalizes to directed graphs, with the same proof: Suppose G is a directed graph such that (1) there is a path between each pair of its vertices, and (2) the indegree of each vertex is equal to its outdegree. Then G has a closed Eulerian path, i.e., a path using each of its edges exactly once (including loop edges, if present). Note that both conditions hold for our graph G_2. (Condition (1) is easy to check, and the indegree and outdegree of each vertex is 2.

(a) Find a closed Eulerian path in G_2. To make your path clear, draw the vertices of the path along with the label edges, in order in a straight line as follows:



The first and last vertices of your path will be the same.

(b) Below, we have arranged 0s and 1s around a circle:



Notice that each binary string of length 2, i.e., 00, 01, 11, and 10 appear as we travel clockwise around the circle. Use your Eulerian path from the previous part of this problem to create a similar circular list of 0s and 1s in which each of the 8 binary strings of length 3 appear. (The edge labels in G_2 are relevant!)