I have read and agree to the collaboration policy. Lynne Diep.

Lynne Diep
Homework Heavy Grading
Collaborators: none

Homework 4-2
Due: June 5, 2017

Runtime	Algorithm
m	Covert G to G', where G' is the bipartite graph - Left side of G' is all vertices in G - Right side of G' is all in G for all v in G - Connect head vertex of G to L to tail vertex in G in R
nm	Run the max-cardinality matching on the bipartite graph in G' (algorithm presented during class) - check is the max-cardinality matching is a perfect matching - L' = nodes that have a match in the max-cardinality matching in L - R' = nodes that have a match in the max-cardinality matching in R
m	- if |L'| == |R'| then the max cardinality matching is a perfect matching. So return edges L' → R' - else return "no cycle cover exists"

Runtime – O(2m +nm)
= O(mn)
Space – O(m + n)

Proof of Correctness:
Claim – A graph has a cycle cover if a max-cardinality perfect matching exists.
The proof of bipartite mac-cardinality matching was provided in class

1. If |L'| == |R'|, it is necessary and sufficient to be a perfect matching
   - Graph G has a subset of x ≤ L'
     Let y be all the nodes adjacent to x in R'
     |y| < |L'| → a perfect matching

2. Perfect matching is the cycle cover.
   3 properties of cycle cover:
   1. Covers all vertices in G
      → by definition of perfect matching, all vertices are matched
   2. Vertices are only in one cycle
      → by perfect matching definition, every node only connects to one other node (other than itself)
      → can only be in 1 cycle
   3. All edges in perfect matching are all edges in a cycle
      → From the perfect matching graph, each node has exactly 1 edge going in, and 1 edge leaving it.
      → Since the property applies to all vertices, Graph G is a graph of cycles, therefore a cycle cover.

Thus, a graph has a cycle cover if a max-cardinality perfect matching exists