HW 4

Due: Fri, 17 Feb 2023

- *Take a look, but do not turn in:*
  1.3.38 Let G be a graph with at least 3 vertices. Prove that G is connected if and only if at least two of the subgraphs obtained by deleting one vertex of G are connected. (Hint: Use Proposition 1.2.29.)

  This problem is potentially confusing: give an incorrect interpretation that could easily be made under exam stress. Then note: (⇒) is easy using the hint, and (⇐) is not hard, but has a subtlety (a special case easy to miss).

1. **Problem 1.3.40.** (!) Let G be an n-vertex simple graph, where $n \geq 2$. Determine the maximum possible number of edges in G under each of the following conditions.

   (a) G has an independent set of size a.

   (b) G has exactly k components.
   The issue here is whether you should have few large K_is or many smaller K_is.

   (c) **skip** G is disconnected.

2. **Problem 1.3.46.** Prove or disprove: Whenever the algorithm of Theorem 1.3.19 is applied to a bipartite graph, it finds the bipartite subgraph with the most edges (the full graph).

   We did Theorem 1.3.19 in class, but there was some confusion about why/how you could switch a vertex v from one side of the bipartition to the other in order to increase $\deg(v)$*. The algorithm does this operation until it achieves $e(H) \geq e(G)/2$ (where H is the bipartite subgraph we are building).*
   At any rate, make sure you understand this algorithm (future-quiz threat ☺).

3. In class I gave the characterization of trees: Let T be a graph with n vertices. Then the following statements are equivalent.

   1. T is a tree.
   2. T contains no cycles and has $n-1$ edges.
   3. T is connected and has $n-1$ edges.
   4. T is connected, and every edge is a cut-edge.
   5. Any two vertices of T are connected by exactly one path.
   6. T contains no cycles, and for any new edge e, the graph $T+e$ has exactly one cycle.

   Prove (5) ⇒ (6).

4. **Problem 2.1.4.** (-) Prove or disprove: Every graph with fewer edges than vertices has a component that is a tree.

5. **Problem 2.1.18.** (!) Prove that every tree with maximum degree $\Delta > 1$ has at least Δ vertices of degree 1. Show that this is best possible by constructing an n-vertex tree with exactly Δ leaves, for each choice of n, Δ, with $n > \Delta \geq 2$.

Bonus problems . . . .

6. **1.3.52.** Prove that every n-vertex triangle-free simple graph with the maximum number of edges is isomorphic to $K_{\lfloor n/2 \rfloor, \lceil n/2 \rceil}$. (Hint: Strengthen the proof of Theorem 1.3.23.)

   We talked about 1.3.23 in class, so you would need to understand that proof in order to follow the hint.

7. Code up the algorithm for Prüfer encoding and decoding. You can use whatever data structure you want to represent trees: edge list, adjacency matrix, . . .