**Homework Five**, for Fri 10/30 CSE 101

Prepare a PDF file in which your solution to each of the following problems (1–5) begins on a fresh page. Upload the file to Gradescope, using your campus email address as login. The deadline is noon on Friday.

These problems cover the following skills and concepts:

- Understanding basic properties of trees
- Reasoning about trees
- Familiarity with Kruskal's algorithm for minimum spanning tree
- The cut property as justification for greedy algorithms that build an MST one edge at a time

---

1. *Kruskal's algorithm and the cut property.* Textbook problem 5.1.
2. *Number of edges in a connected graph.* Textbook problem 5.4.
3. *Characterization of trees.* Show that any undirected graph $G = (V, E)$ that has no cycles and has $|E| = |V| - 1$ must be a tree.
4. *Another greedy approach to minimum spanning tree.* The *cut property* makes it possible to build minimum spanning trees greedily, starting from an empty graph and adding one edge at a time. A different approach is to start with the original graph and remove edges greedily, one at a time, until an MST remains. A scheme of this second type can be justified by the following property.

   > Pick any cycle in the graph, and let $e$ be the heaviest edge in that cycle. Then there is a minimum spanning tree that does not contain $e$.

   (a) Prove this *cycle property* carefully.

   (b) Use the property to justify the following MST algorithm. The input is an undirected graph $G = (V, E)$ with edge weights $\{w_e\}$.

   ```
   sort the edges according to their weights
   for each edge e ∈ E, in decreasing order of w_e:
      if e is part of a cycle of G:
         G = G − e (that is, remove e from G)
   return G
   ```

   (c) On each iteration, the algorithm must check whether there is a cycle containing a specific edge $e$. Give a linear-time algorithm for this task, and justify its correctness.

   (d) What is the overall time taken by this algorithm, in terms of $|E|$? Explain your answer.
5. *Updating an MST when an edge weight changes.* You have a graph $G = (V, E)$ with edge weights $w(\cdot)$, and someone has already given you a minimum spanning tree $T = (V, E')$ of this graph. Suppose, however, that you now need to *increase* the weight of one particular edge $e$. Does the MST change? If so, show how to compute the new MST in just linear time. You should consider two cases: when $e \notin E'$ and when $e \in E'$.