Today's topic: union-find, minimum spanning trees; §§ 24.1...
Next class: nearest common ancestors, more union-find §§ 24.*

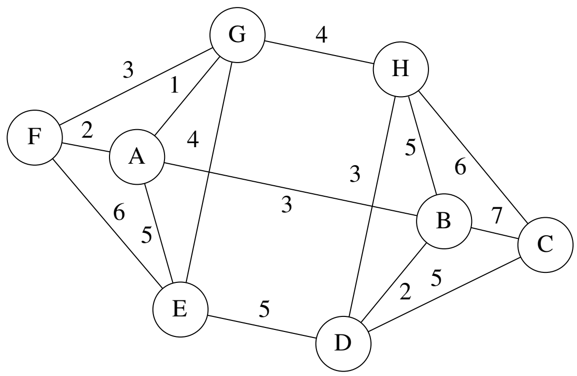
1. List the members of your group below. Underline your name.

2. Provide an illustrative example of an equivalence relation R by specifying the base set S of elements and the pairs $x, y \in S$ for which xRy is true. Do not reuse the examples in the textbook.

   Repeat for relations R_1, R_2, and R_3 which are not equivalence relations because they violate the requirements of reflexivity, symmetry, and transitivity, respectively (with each R_i satisfying the other two requirements).

3. Depict the tree resulting from *find*(7) applied to the tree of Figure 24.17 when path compression is in use. How many nodes get their parents changed?

4. Trace the operation of Kruskal's minimum spanning tree algorithm on the following graph. Indicate each edge that is examined and whether it is accepted or rejected. Depict, after each edge acceptance, (1) the pairing heap used to organize unexamined edges, (2) the forest of accepted edges forming the partial minimum spanning tree, and (3) the forest of the union-find data structure in both tree and array form. Construct the initial pairing heap by inserting edges in lexicographic order of edge names, where an edge (u, v) is named uv if $u < v$ and vu otherwise. For the union-find data structure, use path compression and union by rank.



[additional space for answering the earlier question]

5. Provide an explicit sequence of operations that yields the tree of Figure 24.17 in the textbook when using the union-by-size smart-union algorithm. Depict the forest before and after the *last two* operations.