CSE 326   Problem Set 5   Due: December 3, 1997

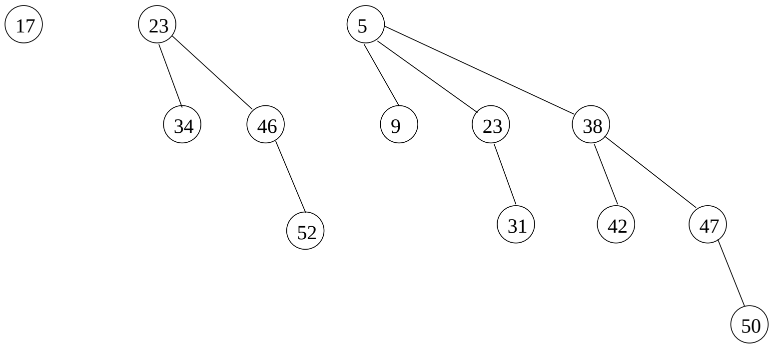
1. This question concerns the amortized analysis of the skew heap algorithm(Weiss, Sec. 11.3). Recall that the *amortized cost* of a step is the actual cost plus the change in potential. Call a binary heap *right leaning* if it has exactly one leaf and each internal node has a right child but no left child, *left leaning* if it has exactly one leaf and each internal node has a left child but no right child, and *perfectly balanced* if all leaves are at the same distance from the root. Determine the amortized cost of merging a skew heap T_1 with n nodes and a skew heap T_2 with m nodes in each of the following cases.

   (a) T_1 and T_2 are both left leaning;

   (b) T_1 and T_2 are both right leaning;

   (c) T_1 and T_2 are both perfectly balanced.

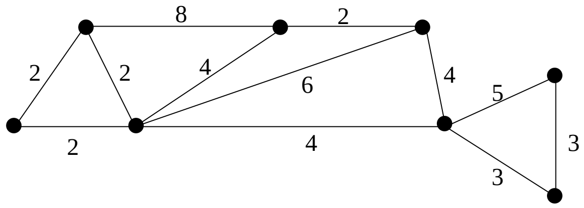
2. Show the result of applying the Delete-Min operation to the following binomial queue.



3. (a) Define the cost of adding 1 to a binary number as the number of bits in the result that differ from the corresponding bits in the original number. In other words, the cost is 1 plus the number of carries in the addition process. Give a tight upper bound on the total cost of starting with 0 and adding 1 n times.

   (b) In terms of O notation, what is the total cost of inserting the integers $1, 2, \cdots, n$ into an initially empty binomial queue?

4. How many minimum-weight spanning trees are there in the following weighted graph?



5. Consider an implementation of the Disjoint Set ADT in which, each set of the partition is represented by a tree of height 1 with the representative of the set at the root, and the size of each set is stored with its representative. This can be done using an array p in which, for each element i that is not the representative of its set, $p[i]$ is equal to the representative of the set containing i, and, for each element i that is a representative, $p[i]$ is equal to a negative number whose absolute value is the number of elements in the set. With this representation the cost of a Find is $O(1)$, but the cost of a Union operation is proportional to the size of the smaller set. Prove that the total cost of starting with n elements, each in a set by itself, and performing n Union operations, is $O(n \log n)$.