CS 561: Homework 9

Prof. Jared Saia, University of New Mexico

Remember: You are encouraged to work on the homework in groups, but please observe the "Star Trek" rule from the syllabus.

1. **Deleting the Larger Half.** Design a data structure that maintains a multiset of numbers, initially empty, and supports the following operations:

   - INSERT(x) inserts x into the multiset;
   - DELETE-LARGER-HALF deletes the largest $\lceil n/2 \rceil$ items, where n is the number of items immediately before the operation.

   If the multiset is empty, DELETE-LARGER-HALF does nothing. You may use a worst-case linear-time selection algorithm as a subroutine.

   Describe your representation and algorithms. Prove that, for any sequence of k operations, the total running time is $O(k)$, not including requests to output the contents. Also explain how to output all n currently stored numbers in $O(n)$ time and how to keep the space usage at $O(n+1)$.

2. **Union by Rank without Path Compression.** Consider the standard tree-based Union-Find data structure using union by rank, but not path compression. Assume that n is a power of two and that $m \geq 3n$.

   Construct a sequence of exactly m MAKE-SET, UNION, and FIND-SET operations that contains exactly n MAKE-SET operations and has total running time $\Theta(m \log n)$. Describe the trees created by your sequence and justify both the number of operations and the running-time bound.

3. **Breaking Every Cycle.** Let $G = (V, E)$ be a connected, undirected graph with nonnegative edge weights. Call a set $F \subseteq E$ a *cycle-breaking set* if every cycle in G contains at least one edge of F. Equivalently, deleting F leaves an acyclic graph.

   Give an efficient algorithm that finds a minimum-weight cycle-breaking set. Prove that your algorithm is correct, and state its running time in terms of $n = |V|$ and $m = |E|$. Hint: Think about what should remain after the edges of F are deleted.

4. **Updating a Minimum Spanning Tree.** Let $G = (V, E)$ be a connected, undirected, weighted graph, and let T be a minimum spanning tree of G. The weight of one edge e changes, while all other edge weights remain the same. Let w' denote the new weight function.

   For each case below, prove the stated update rule. Ties may occur.

   (a) Suppose $e \in T$ and its weight decreases. Prove that T is still a minimum spanning tree under w'.

   (b) Suppose $e \notin T$ and its weight increases. Prove that T is still a minimum spanning tree under w'.

   (c) Suppose $e \notin T$ and its weight decreases. Adding e to T creates a unique cycle C. Let f be a maximum-weight edge of C under w'. Prove that

   $$T' = T \cup \{e\} \setminus \{f\}$$

   is a minimum spanning tree under w'. Notice that f may equal e.

   (d) Suppose $e \in T$ and its weight increases. Removing e from T creates two connected components, with vertex sets S and $V \setminus S$. Let f be a minimum-weight edge under w' crossing the cut $(S, V \setminus S)$. Prove that

   $$T' = T \setminus \{e\} \cup \{f\}$$

   is a minimum spanning tree under w'. Here f may equal e.

   For parts (c) and (d), explain how to find the required edge and construct the updated tree in $O(m + n)$ time.