Homework 2

1. Data structures for disjoint sets

This exercise is about the linked-list representation of disjoint sets with the union-by-size heuristic and the algorithm for computing the connected components of an undirected graph.

(a) During the execution of CONNECTED-COMPONENTS on a graph $G = (V, E)$ with $k$ connected components, where $|V| = n$, $|E| = m$, how many times is FIND-SET called? How many times is UNION called? Express your answers in terms of $n$, $m$ and $k$. What are these counts for the graph in the figure?

(b) Let $A = \{a_1, \ldots, a_n\}$, $B = \{b_1, \ldots, b_n\}$, $C = \{c_1, \ldots, c_n\}$. Deduce a tight asymptotic bound for the time-complexity of CONNECTED-COMPONENTS on the graph $G = (V, E)$, where $V = A \cup B \cup C$ and $E = \cup_{i=1}^{n} \{a_ib_i, a_ic_i, b_ic_i\}$.

(c) Deduce a tight asymptotic bound for the time-complexity of CONNECTED-COMPONENTS on a graph $G = (V, E)$, where $|V| = n$, and $|E| = \lceil 3n^{1.1} \rceil$.

2. Heaps

Illustrate BUILD-HEAP (constructing a max-heap bottom-up) on the array $A = [5, 3, 17, 10, 24, 19, 6, 22, 9, 50]$ (with $n = 10$). Then illustrate an EXTRACT-MAX operation on the resulting heap. For both questions, specify what operations to fix the heap are performed and show the heap after each such operation.

3. Algorithm design

(a) Develop an algorithm that computes the $k$th smallest element of a set of $n$ distinct elements in $O(n + k \log n)$ time. (Hint: Use one of the data structures we have studied.)

(b) Develop an algorithm that computes the smallest three elements and the largest three elements of a set of $n \geq 6$ distinct elements in $O(n)$ time.

Other suggested exercises (for practice; you do not need to turn in)

4. Recurrences

Use induction to show that the following recurrence solves to $T(n) = \Theta(n)$.

$$T(n) = T(\lfloor n/2 \rfloor) + T(\lfloor n/7 \rfloor) + 3n, \quad T(0) = 1.$$
5. Heaps Assume that \(a[1..n]\) is an array implementation of a max-heap.

(a) Is there any known relation between \(a[2]\) and \(a[7]\)? Or between \(a[2]\) and \(a[9]\)? Why or why not?

(b) Let \(b[1..n]\) be the array obtained by reversing \(a[1..n]\). That is, \(b[i] = a[n - i + 1]\), \(i = 1, \ldots, n\). Prove or disprove: \(b[1..n]\) represents a min-heap.

(c) Let \(a[1..n]\) and \(b[1..n]\) represent binary (maximum) heaps. Let \(c[1..n]\) be sum of \(a\) and \(b\) given by \(c[i] = a[i] + b[i]\), \(i = 1, \ldots, n\). Does \(c\) represent a binary (maximum) heap? Justify your answer.

6. Postfix notation First convert the following expression in infix form to postfix form using the stack algorithm discussed in class. Then evaluate the resulting expression using a stack. Show the stack contents after each step in both phases.

\[
E = 2 \ast (3 + 4) + (5 + 6 \ast 2) \ast 4 \ast 5
\]