Basic Algorithms, Problem Set 2
Due February 10 in Recitation

He who learns but does not think is lost. He who thinks but
does not learn is in great danger. – Confucius

1. Illustrate the operation of PARTITION(A,1,12) on the array

\[ A = (13, 18, 9, 5, 12, 8, 7, 4, 11, 2, 6, 10) \]

(You may use either the text’s program or the version given in class,
but please specify which you are using.)

2. Let \( L(n) \), ("L" for lucky) denote the number of comparisons that
quicksort does if each time it is applied the pivot lies in the precise
center of the array. For example, applying quicksort to an array of
length 31, say \( A(1) \cdots A(31) \) objects, there would be 30 comparisons
(between \( A(31) \) and all the other \( A(j) \)) and then \( A(31) \) would end up
in the 16th place and there would be two recursive calls to quicksort
on arrays each of size 15. Find the precise value of \( L(1023) \). (Hint:
that’s one less than 1024!)

3. You wish to sort five elements, denoted \( a, b, c, d, e \). Assume that you
already know that \( a < b, c < d \) and \( a < c \). Write explicitly the 15
possible orders of \( a, b, c, d, e \). Sort (by giving the decision tree) the
elements with 4 further comparisons. (The assumption is actually just
a convenience as giving the full decision tree with 120 nodes would
be exhausting.) (Think of this as a tricky puzzle – and don’t worry
if you don’t get it! One hint: The first question has to split the 15
possibilities \( 8 - 7 \). Why? Can you find the right first question.) ¹

4. Babu² is trying to sort \( a, b, c, d, e \) with seven comparisons. First he
asks “Is \( a < b \)” and the answer is yes. Now he asks “Is \( a < c \)” Argue
that (in worst-case) he will not succeed.

5. Illustrate the operation of COUNTING-SORT with \( k = 6 \) on the array
\[ A = (6, 0, 2, 2, 0, 1, 3, 4, 6, 1, 3) \].

¹This really gives a decision tree for sorting five elements in seven questions. The first
two questions are \( a < b, c < d \) and then compare the smaller elements. By symmetry all
cases look the same.

²former student, now big cheese at GE Hyderabad
6. You are given a Max-Heap with \( n \) entries. Assume all entries are distinct. Your goal is to find the third largest entry. One way would be to \textsc{extract-max} twice and then \textsc{maximum}. How long does this take? Find a better (by which we always mean faster for \( n \) large) way.\(^3\)

Tell me, what do you plan to do with your one wild and precious life? – Mary Oliver, The Summer Day

---

\(^3\)A major goal of this course is to analyze the asymptotic time for an algorithm. As such, you will sometimes be given algorithms to analyze that are not at all optimal – but they should be analyzed for their own sake.