Computing Science 204 (B1)
Merging Sorted Lists
Wednesday, August 2

Example. *Merging Sorted Lists*

Recall that when we use merge-sort to sort a list of n numbers, where $n > 1$, we divide the list into (almost) equal parts of length $\left\lfloor \frac{n}{2} \right\rfloor$ and $\left\lceil \frac{n}{2} \right\rceil$.

Suppose that two ordered lists of size p and q are to be merged as in Mergesort.

(a) Find the *least* number of comparisons necessary before the order of the entire (merged) list is known? Give an example with $p = 2$ and $q = 4$ to show that this result could actually happen.

(b) Find the *most* number of comparisons that must be done before the order of the entire list is known? Give an example with $p = 2$ and $q = 4$ to show that this result could actually happen.

Solution.

(a) The least number of comparisons necessary before the order of the entire list is known is $\min\{p, q\}$, since after this many comparisons, we have run out of elements in the smaller list, and the remaining portion of the larger list is just copied over to the sorted list. An example where this result could actually happen is

$$\begin{array}{ll} p: & 1,\ 2, \\ q: & 3,\ 4,\ 5,\ 6 \end{array}$$

(b) The most number of comparisons that must be done before the order of the entire list is known is $p + q - 1$, and we saw in class that ordered lists that give rise to this result are

$$\begin{array}{ll} p: & 1,\ 3,\ 5,\ 7,\ \ldots,\ 2p-3,\ p+q \\ q: & 2,\ 4,\ 6,\ 8,\ \ldots,\ 2p-2,\ 2p-1,\ 2p,\ \ldots,\ p+q-1 \end{array}$$

For example, with $p = 2$ and $q = 4$, the ordered lists

$$\begin{array}{ll} p: & 1,\ 6 \\ q: & 2,\ 3,\ 4,\ 5 \end{array}$$

require the merge algorithm to make $p + q - 1 = 5$ comparisons.