

sorting ranges of values (ascii or numbers)

Source: <http://coding.derkeiler.com/Archive/General/comp.theory/2005-03/0259.html>

From: Ralph Boland (ralphboland_at_yahoo.com)

Date: 03/26/05

Date: 26 Mar 2005 10:22:32 -0800

I need to sort a range of values
(in my case ascii or unicode characters but in general numbers).

For example, sorting the list:

k d-f t-v n e-i l w p-r a-b q

results in the list:

a-b d-i k-l n p-r t-w.

This can clearly be done $O(N \log(N))$ time
where N is the size of the initial list.

We can also set a lower bound of $\omega(N + K \log(K))$
where K is the size of the output.

I suspect that this problem is $\omega(N \log(N))$, that is independant of K ,
but have not attempted a proof (nor will I).

***** But if anyone knows of a proof or wants to try one,
***** I'd be interested in seeing it.

I am interested in finding a practical algorithm, in particular one that is
faster when there a lots of merges of ranges and works well for small lists.

My current approach is to
merge sorted linked lists until a single linked list remains.
My algorithm applies a heuristic that attempts to ensure that
pairs of lists to be merged are approximately that same size.
I have hadno luck so far in finding a good sorting in place algorithm.
Modifying heapsort looks interesting but complicated.
Quicksort does not look promising.

If anyone knows of good algorithms for this problem or can create one
I be interested in seeing this also.

comp.theory: sorting ranges of values (ascii or numbers)

Thanks

Ralph Boland