35.5 Exercises

Factual

1. If we use counting sort to sort all 61B students by the number of siblings they have, how long will the counting array be? Let N be the total number of students and M be the maximum number of siblings a student has.

2. When performing a radix sort like LSD or MSD, which of the following sorting algorithms could we use for each digit?

   - ☑ Counting sort
   - ☑ Heapsort
   - ☑ Mergesort
   - ☑ Selection sort

☰   C   **CS61B Textbook**   🔍

∨ Problem 1

> The counting array will have M entries. In general for counting sort, the count array's length only needs to be equal to the maximum number of unique elements we are counting.

∨ Problem 2   #

> Recall that the sub-sort for each digit must be stable. This rules out heapsort and selection sort.
>
> - ☑ Counting sort
> - ☑ Heapsort
> - ☑ Mergesort

☑ Selection sort

Problem 3

26, since there are 26 possible values for the first digit we're sorting on.

Metacognitive

1. What are some advantages of counting sort over quicksort?
2. Why do we use LSD over counting sort?
3. If we use MSD radix sort, we start with a single problem of size N, where N is the number of strings. Depending on the results on the most significant digit, we end up with a larger number of smaller subproblems, i.e. we're doing some divide and conquer. In the worst case, we end up with only one subproblem, also of size N. When does this happen?

Problem 1

Many correct answers, but some advantages include:

- For sufficiently large N, counting sort is faster (since it is linear as compared to N log N).
- Counting sort is stable, since we scan from the first element to the last.
- Counting sort doesn't use comparisons between elements, which might be useful if elements are not comparable.

Problem 2

Counting sort is technically faster than LSD, since counting sort doesn't require splitting up items by digits/letters. However, counting sort can require extremely large amounts of memory in some cases (for example, when sorting `String`s, you need one entry in an array for every possible `String`).

Problem 3

This happens when items start with the same digit. In this case, all items end up in the same subgroup after the first iteration of MSD, and we recurse on a problem of size N.

Last updated 1 year ago