

Assignment 4 50 Points

This assignment consists of exercises in analysis of specific algorithms using actual data.

The first three exercises deal with `fsu::List::Sort()` implemented as MergeSort and the two measures of runtime cost:

compares = number of calls to `ValueType::operator<`

advances = number of invocations of an assignment statement `pointer = pointer->next`

1. Argue why *compares* is a measure of the fundamental cost of sorting and *advances* is a measure of overhead cost associated with implementing the algorithm on a linked list structure.
2. Argue that $compares = O(n \log n)$ and $advances = O(n \log n)$. Can you estimate the relative sizes of these two measures?
3. Use the “spy” version of `fsu::List::Sort()` to collect data on key *compares* and pointer *advances* for sorts on various data sets. Then use the collected data and the method of least squares to find best-fit curves for the data, using the form:

$$F(n) = A + Bn \log n$$

There will of course be two different solutions for the two data sets (one for *compares* and one for *advances*)

The fourth exercise asks for an analytical discussion of the possible effects of hash function choices on the runtime efficiency of `fsu::HashTable<K,D,H>`.

4. Provide a narrative analysis of the efficiency of `fsu::HashTable` operations `Insert(K,D)`, `Remove(K)`, and `Retrieve(K, D&)` based on experimentation with various table data files and the `Analysis()` feature. The variable in this analysis is the hash function. Draw conclusions suitable for an executive summary and recommendations.

Assemble your solution paper as follows:

- For each question, repeat the question on the paper, and then provide your solution.
- Take some time to get your math typeset correctly.
- Convert to a pdf document and turn that document in via Blackboard under “Assignment 4”.