How is design of computationally efficient solutions important with respect to economic and social factors. How is efficient computing, and the design of fundamental algorithms for such, aligned with an efficient economy and how can it support the goal for a resource-wise and society wise sustainable future?

Efficiency in coding is important because of time is money. In the context of economics, the faster a program finishes, the more the program can be utilized by its users which promotes the developer’s reputation of creating a quick product. In the context of resource availability, the more space in memory that is saved within algorithm development, the more use the rest of the memory is going to get. Additionally, in a social context efficient code allows ideas to be better shared and collaborated because it is not going in circles or anything. The answer is clear.

Refer to the theoretical complexity of the aforementioned sorting algorithms and relate the expected worst-case and average case with your results in terms of timing the CPU performance. Fit a function of CPU time and problem size and identify if this matches the theoretically expected result.

|  |  |  |
| --- | --- | --- |
|  | Selection Sort | Merge Sort |
| Average Case | O(n^2) | O(*n* log(*n*)) |
| Worst Case | O(n^2) | O(n log(n)) |

Functions based on my data:

This information does not fit the theoretically expected result. This could be due to an error in implementation of the two sorting algorithms.