Micro-benchmarking Sorting Algorithms in C, C#, C++, Java, and Python

C, C++, C#, Java, and Python are five of the most popular programming languages according to measures such as the TIOBE index. These languages have been developed and improved over decades. The purpose of this research is to compare the speeds of current implementations of each language. Micro-benchmarking utilities were created to measure hand-coded and built-in sorting algorithms for each language. Hand-coded algorithms included heapsort, merge sort, quicksort, bitonic sort, and Timsort. The utilities use the algorithms to sort large collections of numeric data, running the sorts multiple times and averaging the results. The utilities will benchmark the algorithms on different data sets, including data sets consisting of integer values and data sets consisting of floating-point values, large data sets and small data sets, data sets with large numbers of repeating values and data sets with completely unique values. The hypothesis is that for the hand-coded algorithms, C or C++ will execute the fastest on most tests, if not all, then Java and C# will follow, and Python, due to the fact that it is the only scripting language in this study, will execute the slowest. The hypothesis for the built-in algorithms is that there will be less of a difference between the execution times than in the hand-coded algorithms, since the built-in algorithms are optimized for each specific language.