Asymptotic Analysis Project

OLS Regression: Sx(n) = a1\*n2 + b1\*nlgn + c1\*n + d1

SelectionSort:

Increasing: S1(n) = 0.0131323\*n2 + 0.001295\*nlgn - 0.00154809\*n + 0.0015.0123

Decreasing: S2(n) = 0.03790475\*n2 - 0.01241592\*nlgn + 0.01456702\*n - 0.01720632

Constant: S3(n) = 0.01328697\*n2 + 0.0000859\*nlgn + 0.0003306\*n – 3.1796

Random: S4(n) = 0.03749846\*n2 - 0.004006\*nlgn - 0.00096829\*n - 0.00357025

InsertionSort:

Increasing: S1(n) = 0\*n2 + 0\*nlgn + 0\*n + 0 = 0

Decreasing: S2(n) = 0.03972097\*n2 + 0.01248085\*nlgn - 0.02394024\*n + 0.02930586

Constant: S3(n) = 0\*n2 + 0\*nlgn + 0\*n + 0 = 0

Random: S4(n) = 0.02863465\*n2 - 0.06277153\*nlgn + 0.07861072\*n - 0.09350338

MergeSort:

Increasing: S1(n) = -0.00005754\*n2 + 0.00068892\*nlgn - 0.00101525\*n + 0.00116642

Decreasing: S2(n) = -0.00005754\*n2 + 0.00068892\*nlgn - 0.00101525\*n + 0.00116642

Constant: S3(n) = -0.00005754\*n2 + 0.00068892\*nlgn - 0.00101525\*n + 0.00116642

Random: S4(n) = 0.00002985\*n2 - 0.00024704\*nlgn + 0.00053903\*n - 0.00069218

QuickSort:

Increasing: S1(n) = 0\*n2 + 0\*nlgn + 0\*n + 0 = 0

Decreasing: S2(n) = 0.00240223\*n2 - 0.00025802\*nlgn + 0.00043518\*n - 0.00094516

Constant: S3(n) = 0.00933230\*n2 + 0.00144575\*nlgn - 0.00171635\*n + 0.00158771

Random: S4(n) = 0.00009899\*n2 - 0.00083048\*nlgn + 0.00124904\*n - 0.00150312

As shown in the above regressions, quicksort performs the best overall because increasing is S(n) = 0 and the rest of the values the low in comparison to the rest of the values. Mergesort and insertionsort perform well when it is constant values, whereas quicksort fails in this area. And insertion sort performs well with with increasing and constant, but not as well in the rest. Selection performs the worst of all.