



Size of string	Average Runtime for Identical Strings (microseconds)	Average Runtime for Random Strings (microseconds)	Average Edit Distance for Random Strings
1	26.694	6.625	0.753
2	25.500	0.743	1.493
4	8.387	7.692	2.879
8	9.997	1.828	5.369
16	21.239	3.905	10.017
32	50.330	9.147	19.007
64	36.270	70.829	36.389
128	598.908	175.008	70.565
256	1209.217	946.689	138.160
512	3476.661	3570.634	272.249

My implementation of the edit distance algorithm starts to take a noticeable amount of time to complete around when the size of the strings is 128. Naturally, strings of size 256 and size 512 take even longer. It is noteworthy that the program is performing either 1000 or 5000 repetitions of the edit distance algorithm for each run, so fact that the completion time is noticeable is certainly related to the number of repetitions as well. There doesn't appear to be a difference in the difficulty between finding the edit distance of a string with itself and finding the edit distance of two random strings. While the two different cases don't produce identical amounts of time, the difference between the runtimes don't seem to be significant, nor do they seem to stay consistent across multiple sizes of strings. It certainly makes sense that there isn't much difference in the average runtimes, as my algorithm creates an $n \times n$ matrix for each of the strings of matching lengths and computes the value of each of the cells in the matrix regardless of how similar the strings are.

The edit distances between the two random words obviously increases as the size of the words increases. It appears that the average edit distances for the random words is a little more than half of the size of the words. This trend holds pretty consistently throughout the different sizes of string lengths.