Benchmark #1

D. Aaron Wisner (daw268)
ECE 5720

Note: All tests were conducted on a desktop running an i7-3770k @ 4.8 GHz with Linux kernel version 4.4.0

Results

	Centroid Method 1 Runtime (Seconds)	Centroid Method 2 Runtime (Seconds)
K = 15	0.342	0.520
K = 17	0.379	0.649
K = 64	1.354	6.540

Centroid Method 1 showed significantly better performance, especially in the k= 64 case, this has to do with C 2d array indexing and the CPU's cache. The C compiler stores the 2d arrays with column elements in the same row adjacent to each other in physical memory. For example x[j][i] is adjacent to x[j][i + 1] in memory. However, x[j +1][i] is not adjacent to x[j][i], rather if x is defined as "float x[M][N];" then x[j +1][i] is a whole 4*N bytes away from x[j][i]. For this reason, it is much more performant to iterate through 2d arrays in a manner where the innermost loop is iterating through the columns. By doing this, it better takes advantage of temporal locality within the CPU's cache, since elements will be accessed sequentially, as opposed to jumping by 4*K bytes each access. This will result in less cache misses and thus better overall performance.