Alex Saalberg

Lab 5

ARM	**no opt**	**opt**
Runtime (matmul)	223.82806914s (18446744072751754083ns)	24.502679067s (1186095543ns)
Cache Misses (matmul)	1,083,789,431	3,554,761
Page Faults (matmul)	3,173	3,173
Runtime (structs)	4.320864291s Init(18446744073048551403ns) comp(566595049ns.)	4.009169214s init(18446744073011240522 ns) comp(264091745ns)
Cache Misses (structs)	2,344,175	3,079,516
Page Faults (structs)	98,386	82,004

INTEL	**no opt**	**opt**
Runtime (matmul)	3.510088633s (2940914202ns)	1.931947513 (1324202191ns)
Cache Misses (matmul)	26,735	10,614
Page Faults (matmul)	1,896	1,896
Runtime (structs)	4.874092764s init(3951719732ns) comp(906058235ns.)	4.349592036 init(3886778198ns) comp(448089566ns)
Cache Misses (structs)	57,531,886	10,261,675
Page Faults (structs)	3,918	4,433

Matmul

When matmul was optimized so that the second matrix used column-order (instead of row-order) arrays it showed a performance increase on both the ARM and Intel systems. Both optimized versions showed a greatly reduced cache-miss count and runtime, but an equal page-fault count.

The arm version's runtime was reduced by a shocking amount: **223** to **24.5** seconds! The intel version's runtime also saw a reduction: from **3.5** to **1.9** seconds. Clearly, the arm version benefited more from this optimization, a fact that is also evident in the cache-miss statistic. Once optimized, the arm version saw a staggering reduction in cache-misses by one billion. (**billion!)** The intel version saw a more modest reduction from **~26,000** to **~10,000**.

The reason why the arm version had a more dramatic cache-miss and runtime reduction is because of the smaller size of the cache on the raspberry pi. The size of the cache on the pi was just right so that there was a huge performance increase when the second matrix was optimized.

Structs

The results of the structs program optimization are different than that of the matmul program.

The results for the optimization on intel were as I expected. The cache misses were reduced (even more than they were for the matmul program) which resulted a lower runtime. This time the page faults increased after the optimization (this was not an anomaly, but confirmed by multiple runs). Evidently, the reduction in cache-misses was enough to offset the effect this may have had. There was about a 50% reduction in the runtime for the comparison section of this program.

The arm version had different results from the intel version. The cache-misses for this version increased after the optimization, while the page-faults decreased. The change in runtime for the comparisons was about the same as that as the intel version, a reduction of about 50%.