

Use perf to explain the different performance behavior between your different implementations of matrix multiplication. Write a report with your findings.

The theory behind performance stats entailing a miss in the L1 cache then leading to it being fetched from the L2 cache which has a lower latency, thus, a quite high L1 miss ratio would be acceptable. In contrast, a miss in the L2 cache on the will lead to a long stall while fetching data from main memory, so only a much lower L2 miss ratio is acceptable.

Based on the performance stats of all 3 different types of matrix multiplication implementations, it seems like based on all 3 results, the tiled implementations had the best performance. This is corroborated by the level of L1 misses versus the L2 miss ratios. The L2 miss ratio seemed comparable for both the blas and loop implementations. Thus, I deduced that the tiled matrix multiplication looked like it did the best to me. Furthermore, the instruction cycles were comparable to blas and tiled where the loop implementation did the worst. However, construct the miss ratios were the most important, hence why I believe that the tiled implementation is the best performing implementation.