Project 1 Experiments

For the basic matrix multiplication trace, I found that a direct mapped cache is the fastest and best choice. I used the parameters (15,7,0) with a victim cache of size 4. Since it is direct-mapped and there is only one block per set, the replacement policy doesn't matter and they will all result in the same AAT. The AAT for this configuration was about 3.87. This was surprising to me because I would have guessed that having some associativity would have improved the time, but every time I added some associativity to the dimensions, the time worsened. With every increased associativity, the time got a little worse and when I tested a fully associative cache, the AAT was significantly worse at about 9.86. I also tested various configurations with and without a victim cache. It was not surprising that every test had a much worse time without a victim cache.

For the tiled version, I tested all the same configurations and found that the exact same configuration of (15,7,0) with a victim cache of size 4 gave the fastest AAT. This configuration is also direct-mapped so the replacement policy does not matter for this configuration either. I also found similar results with the increase of associativity. As the associativity increased, the AAT also increased. The AAT was at its worst when I tested a fully associative cache. The AAT for direct-mapped was just 2.17 compared to 9.75 for a fully associative cache. Another thing that was surprising to me was the effect the replacement policy had on the AAT. The replacement policy doesn't affect direct-mapped cache but had a surprising effect on other configurations. FIFO was often fastest and I expected LRU or LFU to be the fastest.