Arthur Lawson HW1 ReadMe

**Details:**

I ran the code in vagrant box virtual machine on my laptop. These are the numbers that went into my graph:

Bytes Access Time (ns)

1024 1.2207

2048 1.34277

4096 1.34277

8192 1.37329

16384 1.49536

32768 1.51062

65536 1.4801

131072 1.60027

262144 1.65462

524288 2.18773

1048576 5.50008

2097152 6.71923

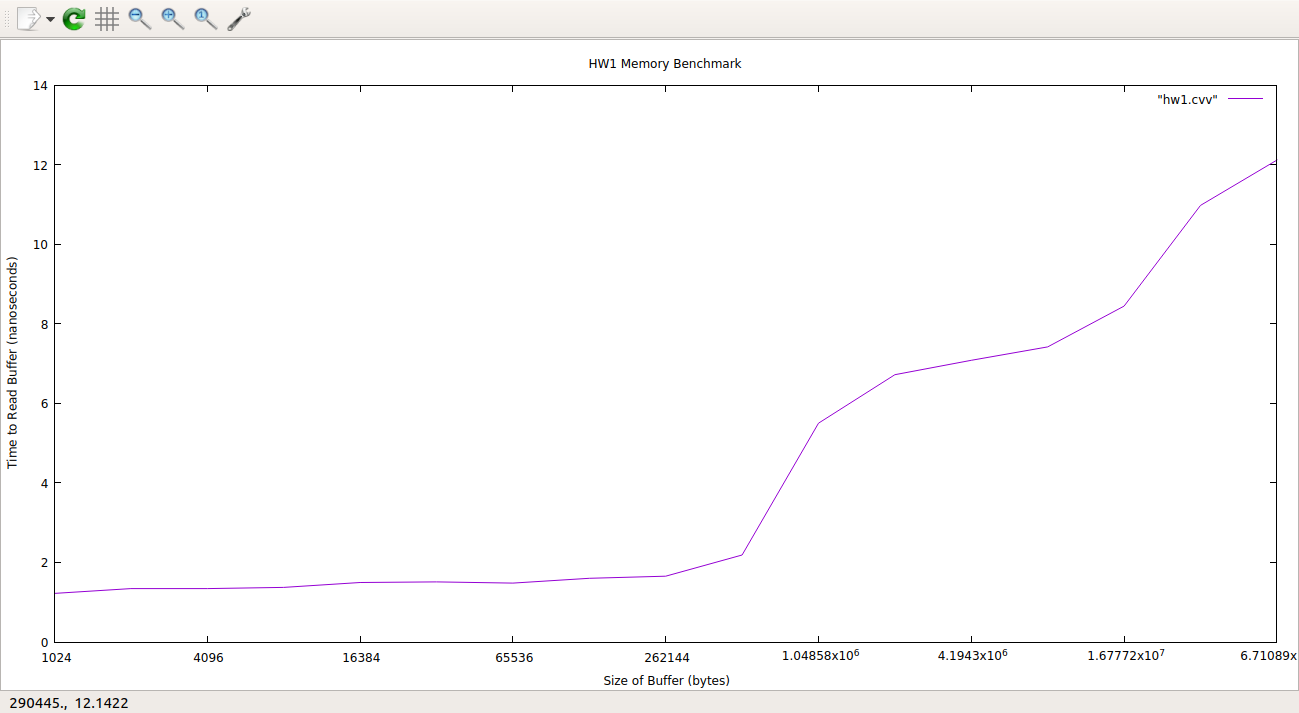
4194304 7.08127

8388608 7.41908

16777216 8.4448

33554432 10.9747

67108864 12.1141



**Graph description:**

In the graph, it is clear that there is a consistent increase in access time as the size of the buffer increases. This is because of the different access times associated with each level of cache, as we have learned in class.

**Part 1:**

Based on the numbers within my graph I believe the L1 cache is approximately 524288 bytes in size because after that iteration, there is a significant jump from approximately 2 nanoseconds in read time to 5 nanoseconds. Following similar logic, I believe L2 is approximately 1048576 bytes in size because the jump from 1048576 to 2097152 was a lot less subtle (nearly a whole nanosecond) when compared to the other jumps within my determined L1 and now determined L2 (which were all less than half a nanosecond). I believe L3 is 6291456 because from buffer size 8388608 to 16777216 there is another jump of a whole nanosecond. Soon after, there is another steep jump (as visible in the graph which is approximately 2 nanoseconds) that shows that the rest is reading from DRAM.

The graph is not completely monotonous. In the L1 there is a small drop, which I attribute to the number of iterations. I believe that if there were enough iterations, this function would be increasing only.

**Part 2:**

The numbers for L1 are in the same ballpark (1-2 ns for me vs 0.5). The L2 cache had a similar jump in size, but still off by about the same (5-6.5 vs 7 ns). My analyzed prediction for DRAM is pretty different from the given data (about 10-12 ns vs 100) which could be error on my analytical behalf, or the prefetcher figuring out what was going on by the time we got to larger byte sizes. Another thing that I have pondered throughout the testing of my code: maybe the buffer sizes are not large enough to require DRAM access and we are just using the caches at this level.

**Part 3:**

According to the linux terminal, the cache size is: 3072 KB which is a little less than half of my alleged L1 + L2 + L3. I feel confident in my assessment of L1 cache, but the analysis for L2 and L3 may not be correctly analyzed.