Jonathan Hart
Program 3 report

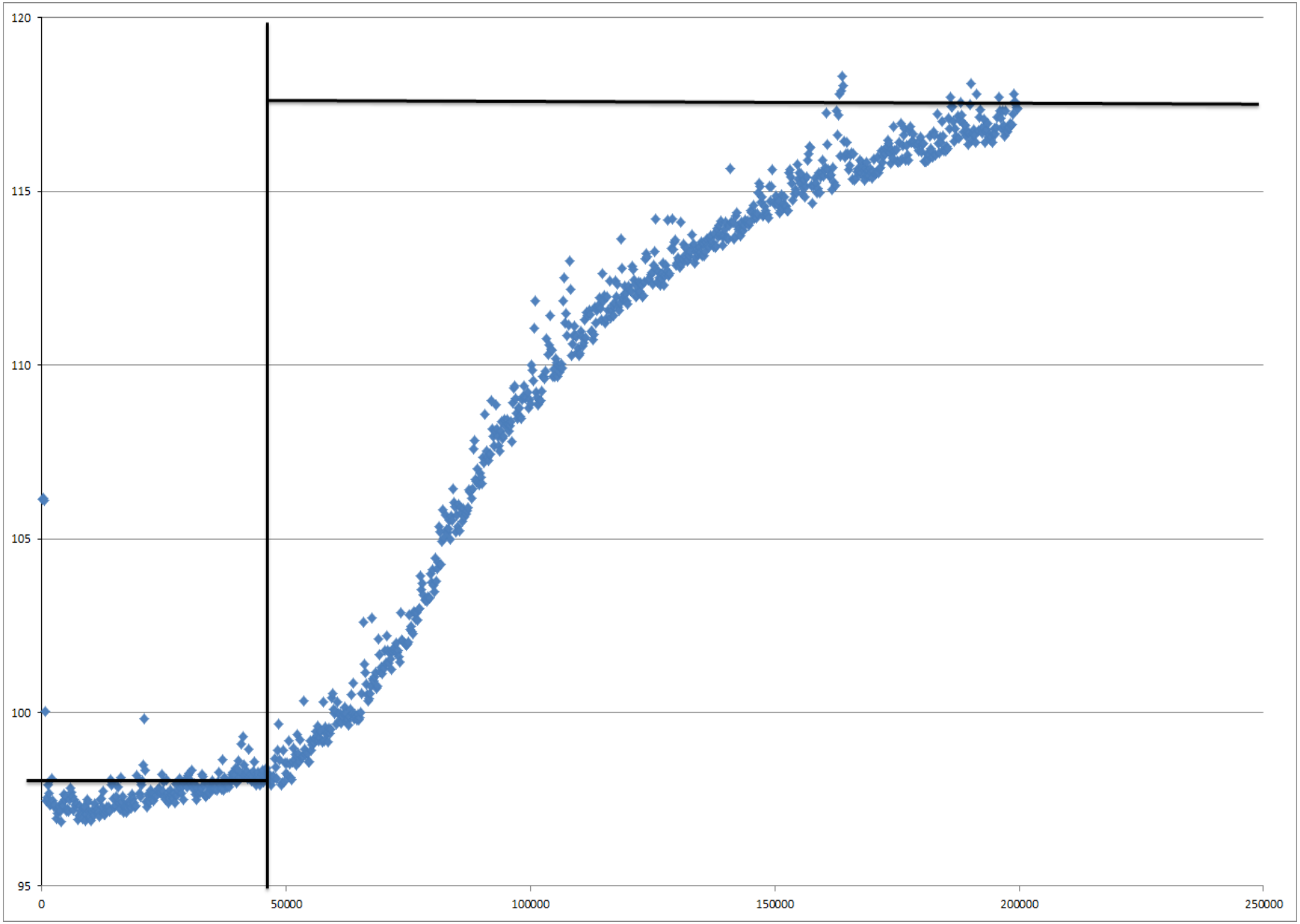
This program I was run completely on Agate during the late night hours, when few, if any people are on. It was attempted on my home windows machine be resulted in some linear results, and due to my own schedule problems, I was unable to test on any of the “lame” university computers. My guess is that the reason for this is because of the Windows operating system and the AMD cpu I use. Here are the lscpu results for agate:

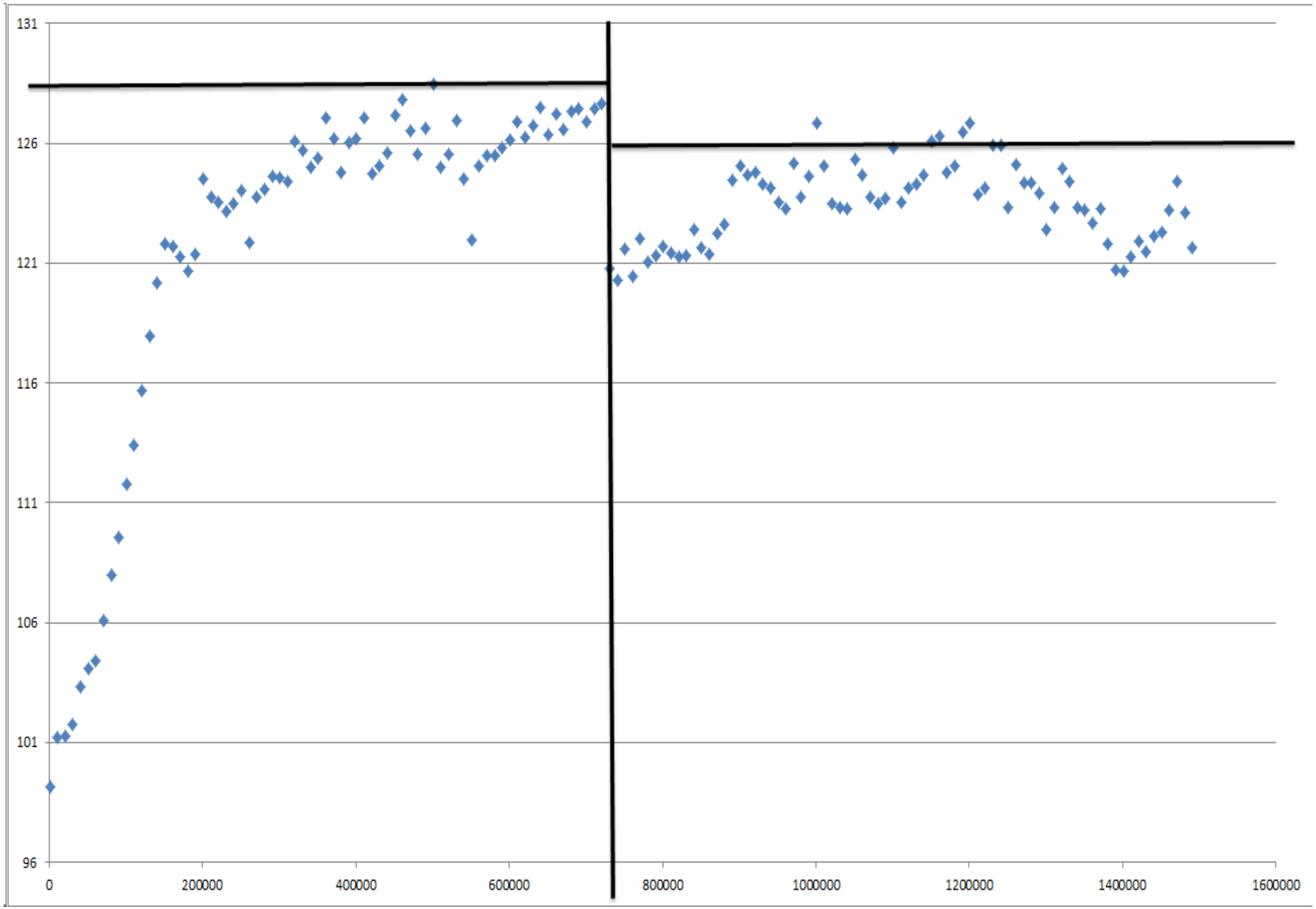
```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                16
On-line CPU(s) list:   0-15
Thread(s) per core:    2
Core(s) per socket:    4
Socket(s):             2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:         44
Stepping:              2
CPU MHz:               3590.933
BogoMIPS:              7181.70
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              12288K
NUMA node0 CPU(s):     0,2,4,6,8,10,12,14
NUMA node1 CPU(s):     1,3,5,7,9,11,13,15
```

The first chart is shown with testing an increasing array size, starting at 200, incrementing by 200 until 200,000 items were in the array. The program timed how long it took to do 10,000 reads. My program ran 50 times and got the average time from the 50 runs. In the chart, the vertical line represents where it shifts from using the L1 cache to using the L2 cache. The horizontal line on the left side represents a rough estimate of the maximum L1 cache time it takes to do the reads and the right horizontal line represents a rough estimate of the maximum L2 cache time.

The second chart is shown with testing an increasing array size, starting at 1000, incrementing by 10000 until 1,500,000 items were in the array. The program timed how long it took to do 10,000 reads. My program ran 50 times and got the average time from the 50 runs. In the chart, the vertical line represents where it shifts from using the L2 cache to using the L3 cache. The horizontal line on the left side represents a rough estimate of the maximum L2 cache time it takes to do the reads and the right horizontal line represents a rough estimate of the maximum L3 cache time.

Clearly agate is too powerful of a machine to get better results to show the different speeds for the caches. For example, the maximum time for the L2 cache is less than the L3 cache. Any timing issues could have resulted from other people access agate but I attempted this during a time when it was likely to be unused. Also, this was tested using for loops and outputting to a text file rather than using a scripting language like python to run the program over and over again. The code to test is commented out in the cache.c file.


120
115
110
105
100
95
0
50000
100000
150000
200000
250000


131
126
121
116
111
106
101
96
0
200000
400000
600000
800000
1000000
1200000
1400000
1600000