Mark Eatough
Ryan Park
CS 2420
1:00 – 2:50

VisualVM Reflection Paper

We were interested to see the graphs and data for how a program runs, and what the machine is actually doing as the program is executing. We had a little bit of a hard time parsing out what parts of the data are useful to us, and what all to look for. It is easy to see how it would be useful to be able to see.

It's interesting to see what we discovered about the graphs and memory use and such. For example, while we were testing the sort test program, we noticed that the integer class is taking up most of the memory as well as the integer array. Meanwhile the other classes and methods were not being used as much, which is what we would expect as the purpose of the program is to sort 1,000 integers.

When we ran our own program which ran from a menu, most of the CPU usage was spent on waiting for us to select a menu option. When we did select a menu option we could see that the method was called, and how many milliseconds it took for the method to run with 1,000 milliseconds being a full second.

All of the programs we have written to this point in our programming classes have been small enough that the effect having inefficient methods or a backing store that takes up more space than necessary has on performance has been negligible to the programs that I have written. The goal is to eventually be working as a programmer, and those programs will likely be large enough and/or written for hardware that has a limited amount of memory. While the process of writing the programs to test the efficiency of programming code may seem tedious, and learning how to read the information on the Java VisualVM will undoubtedly take a lot of work, it will be

valuable to be familiar with in the real world where memory space will be a precious commodity.