This assignment involves performance analysis of matrix multiplication code using PAPI. Pick a machine of your choice to run this on; it can be one of the CS machines or any other machine you have an account on that runs PAPI.

In your writeup be sure to mention what machine you are running on, the kernel version (this can be obtained with the `uname -a` command), the PAPI version, and what type of processor the system has (on Linux this can be seen by running `cat /proc/cpuinfo`).

1. Use PAPI to time the results of a matrix-matrix multiply for sizes ranging from 10 to 300. The matrix elements should be non-zero floating point values. You may use an existing matrix-matrix multiply routine from an earlier homework assignment. Plot matrix size versus time, with a logarithmic axis for time.

2. Determine the size of the L1 Data Cache and L2 Cache on your machine (hint: you can use the `ctests/memory` program that comes with PAPI). Explain any impact you see on your part 1 plot from cache size.

3. Use PAPI to gather data on L1 Data Cache Misses and L2 Total Cache misses, also ranging from 10 to 300. Plot both results in one graph, cache misses versus matrix size, with a logarithmic axis for misses. Explain any impact you see from your processor’s cache size.

4. Plot floating point operations versus matrix size (again from 10 to 300) using the PAPI_FP_OPS event. Are the resulting counts what you would expect given the number of floating point operations you specify in your code? Explain any differences you might find.

5. Based on the results gathered, explain how you might further optimize your code.

Please submit a document with all the graphs produced and any analysis. Also include the source code for your matrix multiply routine that is instrumented with PAPI code.