

Homework 1 assigned. Due Sept. 15 2017, 11:59pm MT.

Dear class,

Below is your first assignment. I'll present it during tomorrow's lecture.

\* Disclaimer:

This first assignment is meant to introduce you to performance measurements for a simple program. Purportedly, you are asked to figure out many things by yourself (how to time a program, how to optimize it, etc.). There are ample references on internet to help you with this assignment, use them.

\* Assignment:

1) Compile and run a C program.

Find a machine (your laptop, a lab machine, etc.) running linux, find out which C compiler(s) are installed (usually, GCC is installed at least), compile and run the basic C program for matrix-multiplication attached: dgemm.c. The output of the execution should be "All done."

In your report, provide:

- which compiler version you used
- the exact command line you used to compile the program

2) Time a C program.

On the same machine/setup, find a way to measure precisely the time taken to execute the function `dgemm_kernel()`, make any modifications necessary to the C file to measure the time taken by this function when executing the program.

In your report, provide:

- where you find the information about how to time a piece of code in a C program
- the exact changes you made (copy-paste the code changes)
- the time taken by `dgemm_kernel()` in the program provided
- the GigaFlop/s achieved by this function

### 3) Optimize a C program.

On the same machine/setup, find a way to improve the performance of the function `dgemm_kernel()`. There is no improvement objective, but you need to improve (reduce) the execution time of the program, by any means you see fit.

In your report, provide:

- an explanation of what you did, and why (1-2 paragraphs)
- the time taken by `dgemm_kernel()` in your optimized program
- the GigaFlop/s achieved by this function

In addition, provide the modified C file for this optimized implementation.

### 4) Analyze the performance of a program.

On the same machine/setup, return to the original C program, and measure the execution time for different scenarios:  $N=512$ ,  $N=1000$ ,  $N=2014$ ,  $N=2000$ ,  $N=2048$  (change the `#define` at the top of the file, recompile and run for each value of  $N$ ). Repeat each run 10 times, to get a total of 50 measurements.

In your report, provide:

- the time taken by `dgemm_kernel()` in the base program, for each value of  $N$ , reporting the min, max and average time across the 10 runs for a particular value of  $N$
- the best GigaFlop/s achieved, for each value of  $N$

(for extra credits, provide also:)

- tentative explanation(s) (1-2 paragraphs) to why there are time variations between runs for the same value of  $N$
- whether the program was SIMD-vectorized by the compiler

\* Submission:

You need to submit, by email to [pouchet@colostate.edu](mailto:pouchet@colostate.edu), subject "CS560 - assignment 1", a PDF with the answers asked above, and a C file with the optimized code.

The deadline is Friday Sept. 15, 2017, 11:59pm MT. No late submission will be accepted (strict).

\* Grading:

This assignment has a weight of 5% towards your final grade.

\* Policies:

This assignment should be completed by yourself, without help from other students. Any material online can be used to complete the assignment.

\* Guidelines:

- It is easier to test programs and time them on linux than windows
- Try to not spend more than 3 hours on this assignment, but you can of course spend as much time as you want/need.
- Be smart: reuse tools, online resources, etc. to get to the solution.