

Objectives: To experience the efficiency of the memory hierarchy.

Activity 1: Utilize a benchmark program in C containing loops to time and print out the MFLOP rating for the following “daxpy” computation:

$$y[k] = a * x[k] + y[k], \text{ where}$$

- $x[]$ and $y[]$ are double precision static arrays,
- a is a double precision constant, and
- k is an integer index arranged for unit strided memory access: $k = i$ where i is the loop counter.

I want you to repeat the Assignment 1 exercise described at:

<http://users.wmin.ac.uk/~lancasd/CST605/assignment.html>

on the Wright hall lab computers under Linux. You can use the program provided, but you will probably need to increase the size the loop ranges. Use the compiler optimization flags: `gcc -O3 daxpy.c`

Model your analysis of the Wright hall computers after the “results” described at (<http://users.wmin.ac.uk/~lancasd/CST605/Assignments/A1/Conclusions.html>) for the data they were provided.

Thus, the steps I want you to perform:

- 1) Read the Assignment 1 description and results on the web
- 2) Use the code provided in the results section to do timings on the Wright hall lab machines
- 3) Perform a similar analysis of the times for the Wright hall lab machines as was done for their machines. Include in your analysis:
 - A printout of the code high-lighting any changes that you made .
 - A printout of the output of your code as columns of size (bytes), time (sec), performance (MFLOPS).
 - A graph displaying the measurement results for the Wright Hall lab machines. The graph should follow the same rules as theirs:
 - The graph should have a logarithmic horizontal axis in “Bytes”.
 - The vertical axis should be in “MFLOPS”.
 - There should be one graph, showing both the unit strided and random sets of measurements.
- 4) Answers to the following questions:
 - a) Estimate (specify the parameters you look up or guess) the peak performance for daxpy on the Wright hall lab machines. How does it compare with the best performance you measure?
 - b) Estimate the size(s) of the cache(s)? Explain your method.

Submit your report by Monday, April 28, 2008.