Due Thursday March 15. No late assignments accepted.

The purpose of this programming assignment is to evaluate the performance of several design alternatives for enhancement of a cache system. Everyone will begin with the same common basic cache system (for debugging purposes).

**Basic Cache Design**

* 256K L1 cache  
* No L2 cache  
* 16 byte block size  
* 2-way set associative  
* lowest numbered cache block replacement policy  
* write through  
* L1 memory access time of 5 clock cycles  
* Main memory takes 40 clock cycles of overhead per access and then delivers 16 bytes every 2 clock cycles (thus it can supply 16 bytes in 42 cycles, 32 bytes in 44 cycles, etc).

You will conduct simulation experiments mimicking the caching behavior of several SPEC programs (we will provide the memory traces for you). The two performance metrics you will compute are **miss rate** and **average memory access time**. You should compare your results with classmates’ to validate your understanding and coding of the basic cache design options.

**Enhanced Cache Design**

You will modify the basic machine to look more like the design of a realistic cache architecture to see the impact on performance. You must choose at least two of the 5 enhancements listed below to add to your simulator.

* Add L2 cache. The total amount of memory in L1 and L2 cannot exceed 2048K but you can distribute them between L1 and L2 as you wish (under the constraint that you have two caches.) Assume L2 access time is 15 cycles, and transfer between L1 and L2 is 16 bytes every clock cycle. Analyze the performance of the options you investigate.  
* Cache block size: 16, 32, 64, 128, 256, 512  
* Set associativity 1, 2, 4 and 8  
* Block replacement algorithm: random v. LRU v. LFU (least frequently used)  
* write back versus write through (and write allocate v. write around) v. writeback
Trace Data. The trace data is located at /cs/classes/www/07W/cis429/MEM-TRACES. There are both READ only traces and READ+WRITE traces there. The formats differ so please read the instructions in README carefully.

Writeup results.
You are to write up your results in the form of a performance evaluation study giving miss rate and AMAT for the various combinations of factors that you have experimented with, and highlighting the best result. Your report should include the following parts:

* One short paragraph describing the goal of the simulations.
* Part I: Basic Cache Design
  o List the parameters assumed (as spelled out above) and define the metrics you compute.
  o Two graphs: one for miss rate per SPEC program, one for AMAT per SPEC program. For consistency, please display them in the order gcc, go, compress95, and perl.
  o One paragraph discussing the results.
* Part II: Enhanced Cache Design
  o List the enhancements that you chose and what range of values you used.
  o Graphs: one for miss rate, one for AMAT with each graph giving results across the SPEC programs. For consistency, please display them in the order gcc, go, compress95, and perl. You should have two graphs for each enhancement separately (across the range of values) and a graph for the combined enhancements.
  o One or two paragraph discussing the results. Do they agree with textbook predictions of trends or not? If not, speculate why not.
* Attach print out of your code.

Be prepared to present your results in class – I will randomly select some people to present and ask you to either email me pdf or bring on a datastick (or bring your own laptop).