CS 530 Fall Semester, 2014
Lab 1: Cache memory
Due 24 October 2014

In this project you will write a cache simulator. You may work alone or in pairs.

Your grade for the project will be equal parts on the code, your documentation, and your testing. You should turn in a report about your code, how it works, how it was tested, and what you learned. Your code should be included with the report and it should be well structured and documented code. You should include tests you used with your code. Please include all you are turning in for the project as a single tar or zip file that you upload to Blackboard.

You will develop your cache simulator to operate as with the cache memory we discussed in class and in your book. The cache will be configurable with command line options for each level of cache as follows (you may include default values and make these optional parameters for your program):

- Word size (number of bytes, same for all levels of cache and memory)
- Block size (number of words)
- Number of lines/blocks
- Associativity (1 for direct)
- Hit time (in cycles)
- Miss time (in cycles)
- Write time to memory (you may assume that writes to L1, L2, or to L3 take the same time as reads to the same level of cache)

Your program will be given a series of addresses and read/write flags (from an ASCII file that contains hex addresses with one address, a space, and a "R" or "W" per line). Note the size of each entry will vary based on the word size. Given this input and for the cache as parameterized at the command line, compute the hit/miss rate for each level of cache and the AMAT.

You may assume LRU is used as appropriate. Your simulator should support L1, L2, L3, and main memory, with L2 and L3 optional at the command line.

Your cache should support write through or write back. You may assume write allocate is used.

We will test your code with the test cases you developed as well as with some additional ones I've developed. The TA will meet with each group to discuss your code, your testing, and other details. If you work in pairs, you should each be able to discuss any aspect of the project.

None of the code for this project is to be developed by others (no finding other resources or sharing code). You may use the STL if you program in C++. If you have other code you want to borrow you must check with me first. Violation of this will result in a zero for the project and potential referral to judicial affairs.