Lab #6: Inferring Mystery Cache Geometries

1. Introduction

Jason Mars, Ph.D, is trying to reverse engineer a competitor's microprocessors to discover their cache geometries and has recruited you to help. Instead of running programs on these processors and inferring the cache layout from timing results, you will approximate his work by using a simulator.

2. Instructions

Specifically, each of these "processors" is provided as an object file (.o file) against which you will link your code. See the file **mystery-cache.h** for documentation of the function interface that these object files export. Your job is to fill in the function stubs in **cache-test-skel.c** which, when linked with one of these cache object files, will determine and then output the cache size, associativity, and block size. Some of the provided object files are named with this information (e.g. **cache_64c_2a_16b.o** is a 64 KB capacity, 2-way set-associative cache with 16B blocks) to help you check your work. There are also 3 mystery cache object files, whose parameters you must discover on your own.

You can assume that the mystery caches have sizes that are powers of 2. You cannot assume anything else about the cache parameters except what you can infer from the cache size. Finally, the mystery caches are all pretty realistic in their geometries, so use this fact to sanity check your results.

You will need to complete this assignment on a Linux machine (e.g. on ECI lab computers) with the C standard libraries (e.g. attu). All the files you need are in **lab6** folder. The provided **Makefile** includes a target **cache-test**. To use it, set **TEST_CACHE** to the object file to link against on the command line - i.e. from within the lab6 directory run the command:

make -f Makefile64bit cache-test TEST_CACHE=cache_64c_2a_16b.o

This will create an executable **cache-test** that will run your cache-inference code against the supplied cache object. Run this executable like so:

./cache-test

and it will print the results to the screen.

3. Project Tasks

Complete the 3 functions in **cache-test-skel.c** which have **/* YOUR CODE GOES HERE*/** comments in them.

Additionally, determine the geometry of each of the three mystery caches and list these in a comment, along with your name, at the top of your modified **cache-test-skel.c**.

4. Turn-in

Submit your modified version of **cache-test-skel.c** to the gradescope. The subject needs to be as follows; <first name1_last name1_first name2_last name2_ECE154a_lab6>. This lab can be done in pairs.

5. Acknowledgments

This lab was originally designed by Prof. Luis Ceze**.**