COLBY COLLEGE
Computer Science Department

CS357 **Allen B. Downey**
Spring 2000

Homework 2: Caching

Due: Tuesday 22 February

The purpose of this assignment is to get you to explore the use of memory caches on real machines, and to give you some practice with UNIX utilities.

Get the program and compile it

1. Pick up the "Cache-testing program" from

   `http://rocky.colby.edu/cs357/code/hw2/cache.tar`

   It is a tar file, which means you will have to use the UNIX utility `tar` to unpack it. Read the documentation for `tar` to find out how.

2. Read the code. There are some system calls and things here that we will probably want to use later, so keep this program around for future reference.

3. The program comes with a Makefile that contains information about how to compile the program. The UNIX utility `make` reads the Makefile and performs the appropriate steps. You should read the documentation of `make` and `gcc`, which is the C compiler.

   Compile and run the program. It takes a while to run, but after a few lines you get a sense of what the output looks like. Take a look at the code and figure out what the units are for Size and Stride.

   When you run the program, you will probably want to redirect the output into a file. For example,

   ```
   ./cache > data
   ```

   will put the output in a file named `data`. The reason it says `./cache` is to specify that the executable named `cache` is in *this* directory, which may not be in the current path.

Start poking around the data

1. Use more, vi or emacs to read the data file.
2. You can use `awk` to extract particular lines from the file. For example:

   ```
   awk '$2 == 4096' < data
   ```

   will print only the lines where the second column is 4096.

   ```
   awk '$2 == 4096 {print $4}' < data
   ```

   selects the same lines as the previous command, but instead of printing the whole line, it only prints the fourth column.
3. You can pipe the output of one command into another. For example

   ```
   awk '$2 == 4096 {print $4}' < data | wc
   ```

   Counts the number of lines (and words and characters) that have 4096 in the second column.

Use gnuplot to plot some data

1. We have a program called `gnuplot` that does some simple plotting. You may find it useful for exploring this data set. If you have another graphing program you prefer, though, feel free to use it instead. There is lots of info about gnuplot at

   `http://www.cs.dartmouth.edu/gnuplot_info.html`

Figure out the cache system

1. Make whatever plots you need to infer the cache size and the block size of the cache. If you can, try to infer the replacement policy (associative, direct-mapped, something else?).

What to turn in

I would like a single report from each pair, with both names on it. The report should include whatever graphs you used, but a bunch of graphs do not make a report. You should explain, in the style I suggested in my example:

- What you set out to discover.
- How the experimental apparatus works.
- How, in theory, you expect the data to look, given an idealized model of the system.
- How, in fact, the data look.
- What, if anything, we can infer from the data.
- What anomalies there are in the data (things that deviate from our theoretical expectation), and any explanation you have for them. By the way, it is ok to acknowledge the existence of these anomalies even if you can't explain them.

Report data sparingly. Do not overwhelm me with uninterpreted graphs. Use more words than numbers. Never report a number without units (unless it is truly dimensionless). Label the axes on all graphs. Write on the graphs.