Homework Assignment 2

CIS501 Fall 2005

Due: Tuesday, October 11^{th} at noon.

Instructions: Your solution to this assignment must be type-written, except for the questions that you answer on the provided worksheet. You should submit the code that you write for questions 3 and 4 via blackboard. For these questions, you should only submit the two files we have asked you to change (`my-cache.c` and `cache-test.c`)- you should **NOT** submit the entire SimpleScalar distribution that we have provided you with. If you do not know how to transfer the files from the system you are working on (i.e. `halfdome` or `eniac-l`,), you should consider using a secure FTP program. CETS provides documentation on how to use SFTP programs for Windows:
`http://www.seas.upenn.edu/cets/answers/filezilla.html`
and for Mac:
`http://www.seas.upenn.edu/cets/answers/fetch.html`

Question 1 (15 points):

(a). A 64KB, direct mapped cache has 16 byte blocks. If addresses are 32 bits, how many bits are used the tag, index, and offset in this cache?

(b). How would the address be divided if the cache were 4-way set associative instead?

(c). How many bits is the index for a fully associative cache. Explain your answer.

Question 2 (20 points): An 8 byte, 2-way set associative (using LRU replacement) with 2 byte blocks receives requests for the following addresses (represented in binary):
`0110, 0000, 0010, 0001, 0011, 0100, 1001, 0000, 1010, 1111, 0111`
For each access, determine the address in the cache (after the access), whether each access hits or misses, and the categorization of each miss

under the "3 C" model. Fill in the worksheet at the end of this assignment with your answer to this question (Note that the first access is done for you). You should fill in the cache lines with the tags that reside there, with the most recently used tag first.

Question 3 (30 points): You have recently been hired by Penntel, the worlds largest maker of imaginary microprocessors, to work on the cache for their upcoming Plentium processor. As your first assignment on this new job, you have been asked to determine the cache size and associativity of several existing processors. Instead of asking you to write a program that runs on a real processor and uses timing to infer cache misses, we are instead going to use a simulator. Each of these "processors" is provided to you as an object file (found in `~cis501/SimpleScalar/hwk2/caches`). See `~cis501/SimpleScalar/hwk2/cache.h` for the functions that these object files provide, and documentation about what they do. Your job is to write a program 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 (i.e. `cache_64_2_16.o` is a 64 KB, 2 way set-associative cache with 16 byte blocks) to help you check your work. There are also `mystery` object files, whose paramaters we are not revealing. The `Makefile` provided includes a target `cache-test`. To use it, set `TEST_CACHE` to the object file to link against on the command line- i.e.

```
make cache-test TEST_CACHE=caches/cache_64_2_16.o
```

For this question, you will edit `cache-test.c` and will submit your code electronically. You should fill in the 3 functions which have `/* YOUR CODE GOES HERE */` comments in them.

Question 4 (35 points): Having now studied the caches on several imaginary processors, you are ready to begin running cache simulations on a typical workload. Fortunately for you, workload of imaginary microprocessors is accurately represented by `vpr.route`. For this question, you will write a cache simulator which conforms to the interface in `cache.h` by editing `my-cache.c`. When editing this file, you may add any datastructure, helper functions, or other code as you see fit– but do not change the signatures of the provided skeleton methods. For simplicity, your cache will always be 2 way set associative, and use an LRU replacement algorithm. When you have completed, you can link it with the SimpleScalar distribution provided for this homework. Run the simulator on `vpr.route` for the following cache parameters:

- 64KB, 64 byte blocks (`-cache:size 64 -cache:bsize 64`)
- 64KB, 128 byte blocks (`-cache:size 64 -cache:bsize 128`)
- 128KB, 64 byte blocks (`-cache:size 128 -cache:bsize 64`)
- 128KB, 128 byte blocks (`-cache:size 128 -cache:bsize 128`)

Fill in the table on the worksheet with the cache hit rate for each set of parameters. Assuming that

- A cache hits in a 64 KB cache takes 2 cycles
- A cache hit in a 128 KB cache takes 3 cycles
- A cache miss takes, on average, 11 cycles

Which cache design should Penntel put in the Plentium processor line? For this question, you should also electronically submit your code.

Questions 2 and 4 Worksheet

Homework Assignment 2
CIS501 Fall 2005

Name:

Question 2:

Address	Line 0	Line 1	Hit or Miss type
0110	(empty)	01 / (empty)	Compulsory miss
0000			
0010			
0001			
0011			
0100			
1001			
0000			
1010			
1111			
0111			

Question 4:

Cache size	Block size		
	32 bytes	64 bytes	128 bytes
64 KB			
128 KB			