COMP411006 Computer Architecture, Fall 2018

Project 2: **Cache Simulator**

Instructor: Prof. Seokin Hong (seokin@knu.ac.kr)

Assigned: November 26, 2018

Due: 11:59 pm December 20, 2018

1. Description

The goal of this project is to write a program that simulates the n-way set-associative cache. The simulator you'll implement will read a memory trace file that contains the memory access trace of load and store instructions executed for a program. The simulator should be configured to have different associativity, cache size, block size, and replacement policy. Thus, your simulator needs to support the following parameters.

-s cache size

-a associativity

-b block size

-r replacement policy

-f trace file name

The cache size and block size are in bytes. The size, block size, and associativity are in the power of 2. Cache size will be 8KB ~ 256KB. Block size will be 4B ~ 128B. Associativity will be 2 ~ 64. The simulator should support the LRU (lru) and Random (rand) replacement policy (-r). Assume that the write policy is the write-through and write-allocate.

You can download a skeleton code (cache_sim.c) and the memory trace file from the LMS. You can use the skeleton code to make the simulator.

2. How to run

// Simulate 16-way set-associative cache (size: 32KB, block size: 32B, replacement policy: LRU)

$ cache_sim -s 32768 -b 32 -a 16 -r lru -f memtrace.trc

// Simulate 4-way set-associative cache (size: 8KB, block size: 8B, replacement policy: RANDOM)

$ cache_sim -s 8192 -b 8 -a 4 -r rand -f memtrace.trc

3. Expected output after running your simulator

```
$ cache_sim -s 1024 -b 32 -a 4 -r lru -f memtrace.trc

cache_size: 1024 B
block_size: 32 B
associativity: 4
replacement policy : LRU
cache accesses : ?
cache_hits : ?
cache_misses : ?
cache_miss_rate : ?
```

* Cache accesses = cache hits + cache misses

4. What to turn in

Please upload the **cache_sim.c file** and a **MS WORD file** to the LMS system (project section).

In the MS WORD file, you need include three things.

1) **Brief explanation of your code**

2) **Four graphs** that show the cache miss rates with different cache parameters. For all graphs, y-axis should be the cache miss rate.

   a. ***Graph 1*:** Evaluate the cache miss rates of the simulated cache with different block size.

      - X-axis should be the block sizes (4, 8, 16, 32, 64, and 128)
      - Fixed parameters: Cache size=32KB, Associativity=8, Replacement Policy: LRU

   b. ***Graph 2*:** Evaluate the cache miss rates of the simulated cache with different cache size.

      - X-axis should be the cache sizes (8KB, 16KB, 32KB, 64KB, 128KB, and 256KB)
      - Fixed parameters: Block size=32B, Associativity=8, Replacement Policy: LRU

   c. ***Graph 3*:** Evaluate the cache miss rates of the simulated cache with different associativity

      - X-axis should be the cache sizes (2, 4, 8, 16, 32, and 64)
      - Fixed parameters: Cache size=32KB, Block size=32B, Replacement Policy: LRU

d. ***Graph 4*:** Evaluate the cache miss rates of the simulated cache with different replacement policy

- X-axis should be the replacement policy (Random and LRU)
- Fixed parameters: Cache size=32KB, Block size=32B, Associativity=8

3) **The contribution of each your team member**. You need to specify the contribution between your teammate, say 5:5 (Default), 6:4, or others.

Late Day Policy

This project is due at 11:59pm on the due date. A grading penalty will be applied to late assignments. Any assignment turned in late will be penalized 25% per late day.

Plagiarism

No plagiarism will be tolerated. If the assignment is to be worked on your own, please respect it. If the instructor determines that there are substantial similarities exceeding the likelihood of such an event, he will call the two (or more) students to explain them and possibly to take an immediate test (or assignment, at the discretion of the instructor) to determine the student's abilities related to the offending work.