CS6560 JAN-MAY 2018
Teacher: Prof. Madhu Mutyam
Assignment 1
Cache Simulator

January 24, 2018

1 Introduction

In this assignment, you will evaluate the performance of different applications on different cache hierarchies and with various replacement policies. To achieve the goal, you will build a *Trace-driven cache simulator* and validate the design. Then study the behavior of various applications using the designed cache simulator.

2 Cache Simulator

Design a multi-level cache hierarchy that supports multiple replacement policies and write-policies, Inclusion/Exclusion policy. Extract your own memory traces from different benchmarks in SPEC CPU 2006 benchmark suite. Each memory trace file should contain 1M memory references. Memory reference traces should be collected using Intel's *PIN*. You have to feed these traces into the designed simulator.

2.1 Trace file format

Trace files will be in ASCII format. Each line in the trace file represents a single memory reference (Please refer to sample trace file (*memTrace.dat*) from moodle).
Format: *memoryAddress Read/Write*
memoryAddress refers to byte address of the memory reference. This address is in hexadecimal format, and it is a 64-bit address in the range of [0-0xffffffffffffffff]. *Read/Write* is encoded as following:

0	Read
1	Write

Each team has to submit 2 public memory traces before January 30, 2018.

2.2 Input Format

Your cache simulator should take the following arguments from a input file.

- Number of levels in cache hierarchy
- Inclusion policy
- Block size

- Cache configuration for each level
  - Cache size
  - Associativity
  - Replacement policy
  - Write policy

We will provide a sample config file "*config.txt*". You can test your design by modifying the config file accordingly.

2.3 Output Format

The output file should contain the following parameters.

- Number of memory references (*numRefs*)
- Number of reads (*numReads*)
- Number of writes (*numWrites*)
- Number of Hits (*numHits*)
- Miss ratio (*missRatio*)
- Number of write-backs (*numWritebacks*)
- Number of clean evictions (*numCleanEvicts*)

All the parameters mentioned above should be reported for each of the caches (Please refer to template output file named "*Stats_temp.txt*").
The format in "*Stats_temp.txt*" is as follows: *C1.L1.numRefs*. Here, C1 represents core id (1), L1 = cache level 1. Core id should start from 0 and cache levels should start from 1.

3 Submission Rules

You have to submit a tar file with following name: *P1_RollNumber#1_RollNumber#2.tar.gz*. The folder should contain a makefile, which will build the simulator when we type **make** in the folder name *P1_RollNumber#1_RollNumber#2*. The makefile should generate an executable *cachesim*. Executable should take two input files as command line arguments.
Usage : *./cachesim config.txt trace.dat*
When we run the executable *cachesim*, it should generate an output file named "Stats.txt". You have to submit another file named "Report.txt", where you will mention your contributions (*RollNumber : Contribution*) in the assignment.

3.1 Evaluation Deadlines:

- Release of assignment 1: January 24, 2018
- Submission of Memory Traces: January 30, 2018.
- Mid-term evaluation: February 3, 2018.
- Final Deadline: February 10, 2018.