Set Associative Cache Project

Contributors:

Victor Porter
Frederick Ward

Implementation Description

For our project we used **python 3.** Our project has three classes defining the cache, the Cache class, Set class, and Block cache.

The Cache class represents a set associative cache and is initialized with three parameters: the size of the cache, the number of ways, and the size of the cache line. An object of this class holds all sets in the cache.

The Set class represents an individual set. Each set holds the b cache lines/blocks and a queue for LRU replacement policy.

The Block class represents a cache line. Each block holds the tag, a valid bit, and a the data in the cache line.

Cache Line Identification Policy

The cache line is identified by obtaining the tag, set index, and offset from the virtual address and searching through the set for a cache line with the same tag.

Placement Policy

A new cache line is placed in the first invalid cache line element. If there are no invalid cache lines, the replacement policy is enacted.

Replacement Policy

We used the LRU(Least Recently Used) replacement policy. It is implemented by a queue for each set that holds the most recent accesses.

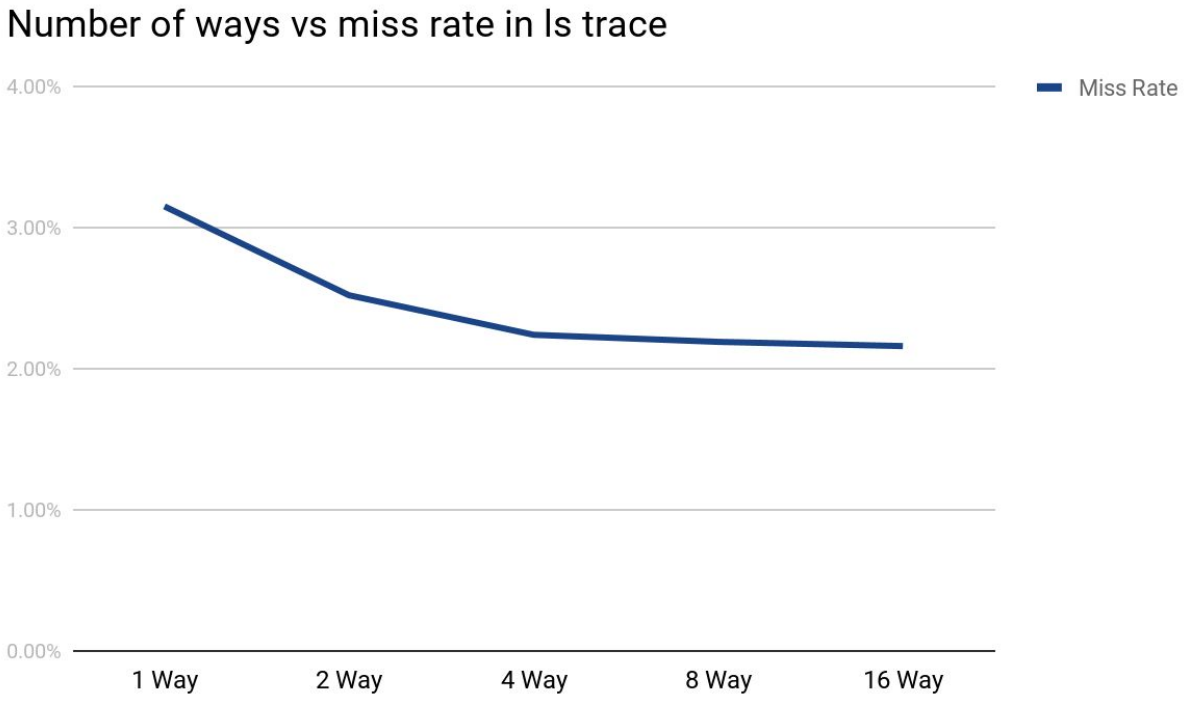
Experiment Description

We began by testing with a python script (input_gen.py) that creates random hex values. When all errors were fixed we moved on to testing the provided files.

Results

Trace File	Configuration (Cache size, Line Size, Ways)	Cache Miss Rate
1KB_64B	1KB, 64B, 16	50%
4MB_4B	4MB, 64B, 16	2.08%
32MB_4B	4MB, 64B, 16	6.25%
bw_mem	4MB, 64B, 16	1.56%
ls	32KB, 64B, 16	2.168%
gcc	32KB, 64B, 16	1.89%
native_dgemm	256KB, 64B, 16	50.24%
native_dgemm_full	256KB, 64B, 16	49.37%
openblas_dgemm	256KB, 64B, 16	8.292%
Openblas_dgemm fulltrace	256KB, 64B, 16	7.49%

Trace File	Configuration (Cache size, Line Size, Ways)	Cache Miss Rate
ls	32KB, 64B, 1 Way	3.15%
ls	32KB, 64B, 2 Way	2.52%
ls	32KB, 64B, 4 Way	2.24%
ls	32KB, 64B, 8 Way	2.19%
ls	32KB, 64B, 16 Way	2.16%



Contribution Description

- Victor Porter - Programming Cache simulation internals, some testing
- Frederick Ward - Programming Main file and Wrapper file, testing