

Ian Buitrago  
Miguel Diaz  
11-30-2011

## Report

### CS 352 project: L1 cache simulator

#### Problem statement:

The goal of the project is to create a level 1 data cache simulator in C++. Using object oriented programming, the source code is written to clearly illustrate how a data cache works. Also, by passing different parameters for the dimensions of the cache/memory, the statistics printed by the simulator can be compared. Varying parameters cache size (KiB), block size (Bytes), and associativity will affect the cache miss rates.

#### Solution:

heuheuheuheu

#### Experiments:

The top 3 optimal configurations are :

#### Class design:

The cache simulator with 4 classes. MainMemory, CacheMemory, Set, and CacheLine. Each aims to simulate that hardware compenent. Most of the implementation is in the CacheMemory class. It contains a MainMemory object and an array of Set objects. It also contains all the statistic variables and other useful variables as private members.

MainMemory has an integer array to hold all the data. It has simple accessors and mutators to interact with that data along with a print() method called by CacheMemory's print().

The Set class contains an array of CacheLine objects, the LRU algorithm, and read, write and print methods that are used only by the CacheMemory object.

The CacheLine contains and array of integer words, the tag and the valid and dirty bits. The print method is only called by Set's print method.

All the constructors help to initialize the data automatically when the objected are created.

