COSC 420- Computer Architecture
Spring semester 2024
Project 1 – Cache Memory Simulation - due date 03/10/2024

Your assignment involves developing and executing a simulator for a modern processor system. This system must include:

- A two-way set associative cache memory of 512 bytes.
- Cache blocks of 64 bytes each.
- An additional "victim's cache" capable of holding two blocks.

The primary cache uses the LRU (Least Recently Used) algorithm for replacing lines. When a block is removed from the main cache, it first goes to the victim's cache before being written back to memory. The victim's cache operates on a FIFO (First In, First Out) basis. A new block entering the victim's cache displaces the oldest block. If an address is needed and not found in the main cache but is in the victim's cache, it's moved back to the main cache (and concurrently removed from the victim's cache). This does not count as a cache miss.

Your software will process the file 'data.txt', which simulates cache requests and calculates the total number of misses. The solution should be specific to this task without options for customization like adjustable cache or block sizes.

The file contains 1 million addresses.
Each record in the file has two parts: a Code and an Address (in hexadecimal).

The Code, a single-digit number, is not used in the simulation:
0: Data read address
1: Data write address
2: Instruction fetch address

The Address is the key information for the simulation.

Example Entries in the Data File:

```
2 415130
0 1010acac
2 415134
```

Addresses are 32-bit. If an address is shorter, it should be zero-padded.

Develop the program in C or C++.
The code should be direct and specific to the problem. Avoid creating a versatile solution for different cache configurations.

Submit your source code and the total number of misses. The execution results can be included as a comment in the source code or in a separate file. This project is to be completed individually.

Note: My implementation of the simulator with the 'data.txt' file resulted in 124,870 cache misses.