CSCE230302 - Comp Org. and Assmbly Lang Prog

Project 2: Memory Hierarchy Simulator

Professor: Dr. Cherif Salama

Semester: Fall 2021

Brief description and bonus feature:

The simulation allows direct, fully associative, and n-set associative mapping where n is specified by the user.

Assumption:

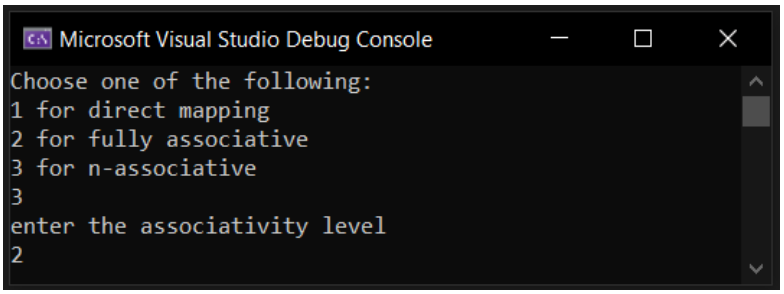
The only assumption to mention here is that in the fully associative mapping and n-set mapping, we considered random replacement once the blocks and sets are full (fully associative and n-set associative respectively).

Any bugs or issues:

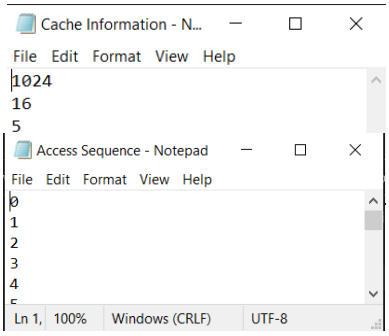
No bugs or issues were detected, and we tested a lot of tests, including different possibilities for misses and hits and it always produced the expected results.

User Guide:

The simulation starts by allowing the user to choose one of the mapping ways, and if n-set associative, the user then specifies the associativity level. It then reads the inputs files that the user write as follows:



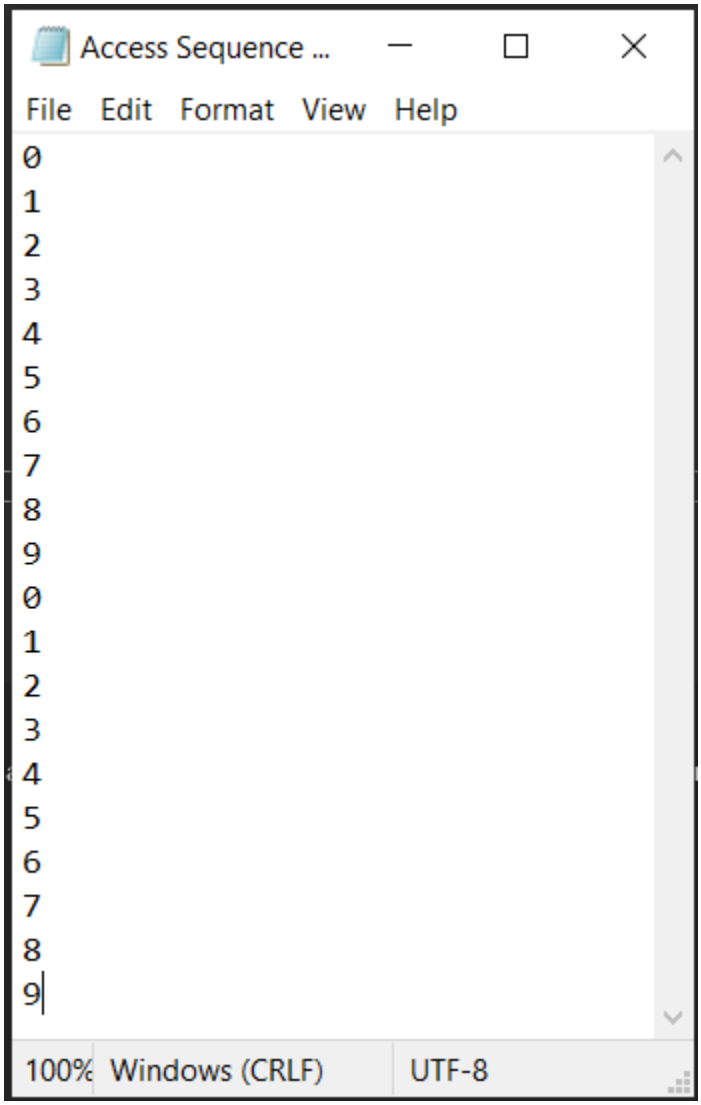
1. Cache Information:
    a. Line 1: The total cache size S
    b. Line 2: The cache line size L
    c. Line 3: The number of cycles needed to access the cache
2. Access Sequence: A sequence of memory addresses that are accessed by a certain program. Addresses should be given in bytes. Each address is written in one line.

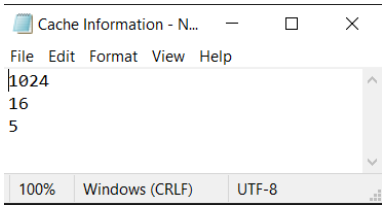


Test Cases:

Test cases were designed to test different possibilities of misses and hits. Input files and results will be shown for each test in this section, and they can be found here as well.

1. This one was designed to make all accesses are hit (except for missing the block in the first time only).





As expected, it produced one miss at the very beginning, then all of the following were hits, and it showed the same results in direct mapping, fully associative, and set associative mapping. For the set associative, we tested with n=1 to check if it is the same as direct mapping or not (which is the case) and with n = 2 for variety.

Results:

Direct mapping:
https://drive.google.com/file/d/1mfzAULQGyecLuyYWkSy9K6J7vEKxllp_/view?usp=sharing

1-set associative:

https://drive.google.com/file/d/1pgrJC16z_XEbAWLT6m_jHUdMMmTLKVYU/view?usp=sharing

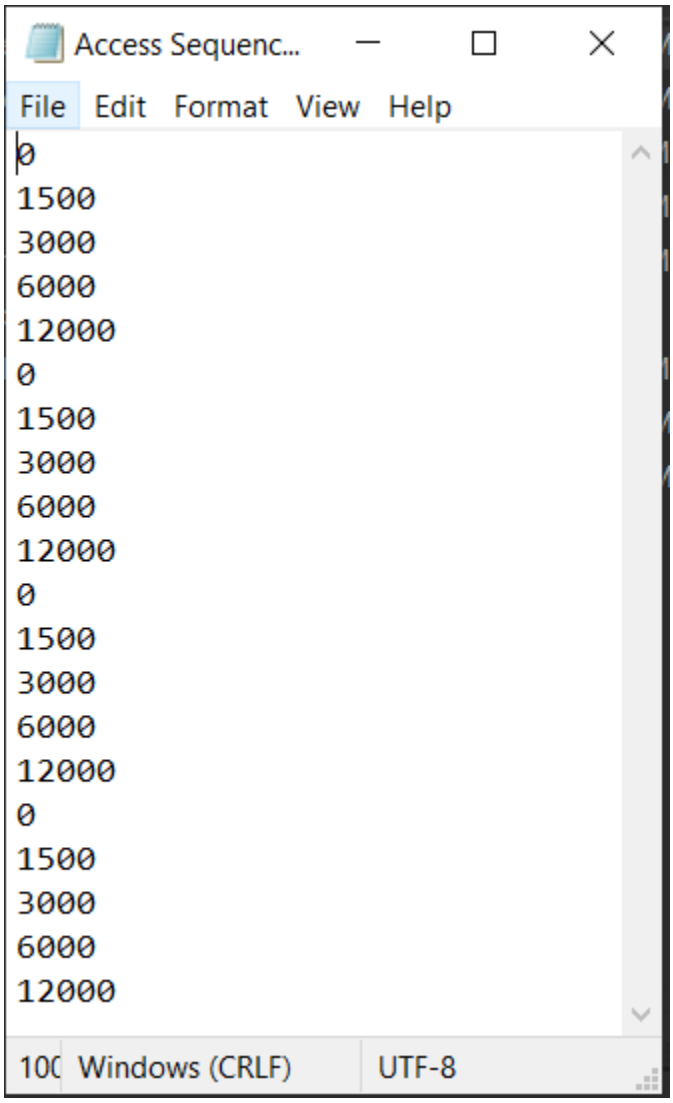
2-set associative:

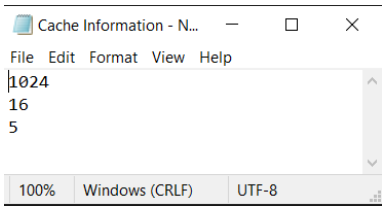
https://drive.google.com/file/d/1OR_m9sSf2kB1zaJQYOdrmAbOqMqHaimf/view?usp=sharing

Fully associative:

https://drive.google.com/file/d/15gsek1bDBtNTuYgIq8PHJbZMwSOCMRc3/view?usp=sharing

2. It was designed to test access of 5 random addresses with repeated for four times (chosen arbitrarily) as follows:





Results:

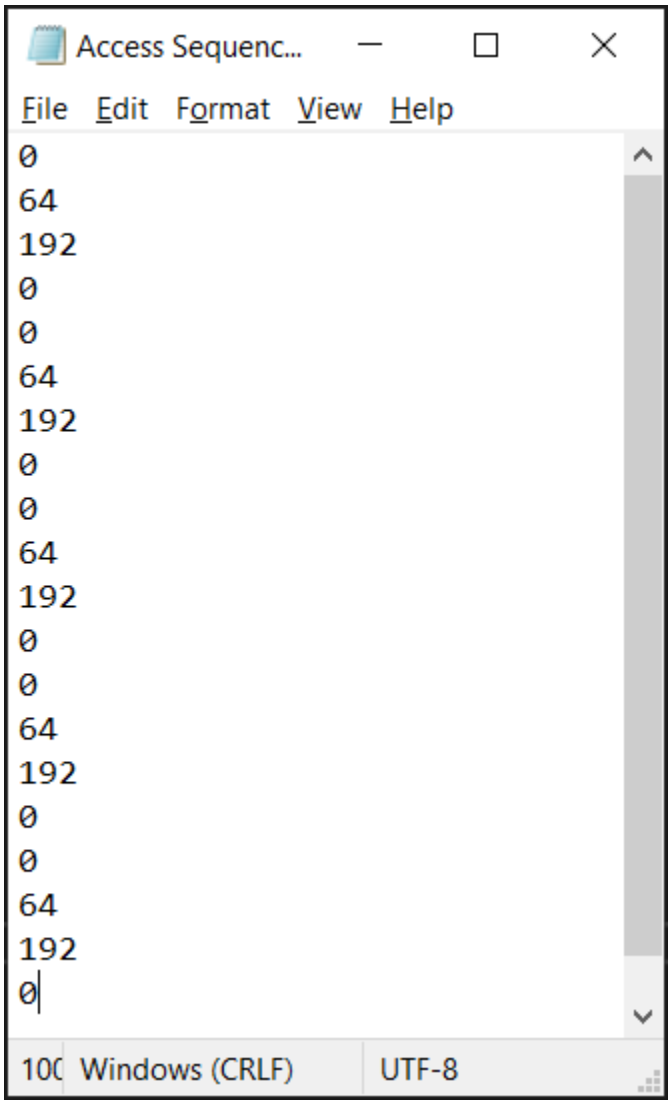
Direct mapping:
https://drive.google.com/file/d/1di3wm6mVPVYtDnle5zjvDBfmaZKJUESY/view?usp=sharing

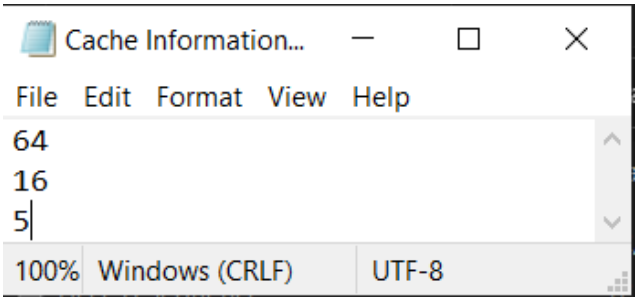
1-set associative:
https://drive.google.com/file/d/1YznTMvPocouctMUCb1siKj6z27IBFOYr/view?usp=sharing

2-set associative: https://drive.google.com/file/d/13_lvMORPdfO9ZHhdCUOqMvB4VM0-GNma/view?usp=sharing

Fully associative:
https://drive.google.com/file/d/12bLfhje7YwJoDEzoQFyz7bPA3noifHj6/view?usp=sharing

3. It was designed to generate a 4-block cache for better analysis and to mainly show:
    a. Direct and fully associative mapping produces different results
    b. Direct mapping and fully associative are special cases of the set associative respectively as follows, so we expect almost same results (not necessarily the same because of the random replacement policy we assumed).
        i. 1-set associative
        ii. n-set associative
    c. Increasing the associativity increases the hit rate





Results:

Direct mapping: https://drive.google.com/file/d/1iMRPeBGrXocovEkF5oiEE-BAI4lM2awt/view?usp=sharing

1-set associative: https://drive.google.com/file/d/1spd3aK4i4_YJx_fCybhHSbrXZlqbPoxJ/view?usp=sharing

2-set associative: https://drive.google.com/file/d/1eiHeRA0RygWiS9N-mxsKVAj-1mYYgAET/view?usp=sharing

4-set associative: https://drive.google.com/file/d/1vniEo0HpWFvCGgqxDCfw14PrHRzKfa0l/view?usp=sharing

Fully associative: https://drive.google.com/file/d/1HHupIGL6Fxp2D0Psh_ow7PBohplCN13Z/view?usp=sharing