This was my last assignment for my operating systems class at Sac State.

The objective for this assignment was to simulate three page-replacement policies for virtual memory: First-In First Out (FIFO), Optimal and Least Recently Used (LRU).

- **FIFO:** The first page loaded into a physical frame is selected for unloading (replacement).
- **Optimal:** The page that will not be accessed for the longest time in the future is selected for unloading (replacement).
- **LRU:** The page that has not been accessed for the longest time is selected for unloading (replacement).

To simulate this we will read an input text file and write to an output text file in the same directory as the Assignment4.java file. The input and output files must be named input.txt and output.txt respectively. Sample test values are provided.

Then you need to update the rootFilePath string variable to the same name as the directory your files are stored. Now you are ready to run the program. If you would like to run multiple tests you can enable automated testing. The automated testing was not part of the assignment but something I added to make testing less tedious.

Automated Testing: To use automated testing your test input/output files must be in the format **'testN.txt'** and **'testNo.txt'** respectively. Where 'N' is the test number.

- **Step 1:** Set the **enableAutomatedTesting boolean** to **true**.
- **Step 2:** Set the **testCount int** to the number of test input files you would like to run.
- **Step 3:** Run the program
- **Done:** You can review the test results in the console. Results show output versus expected preceded by an **ERROR** if they don't match. The total errors for all tests displays last.

Example

Here is an example input file and the corresponding output file. The input has 8 pages, 4 frames and 12 requests.

Input:

8 4 12

4

3

4

6

1
6
4
5
2
4
6
1
Output:
FIFO
Page 4 loaded into Frame 0
Page 3 loaded into Frame 1
Page 4 already in Frame 0
Page 6 loaded into Frame 2
Page 1 loaded into Frame 3
Page 6 already in Frame 2
Page 4 already in Frame 0
Page 4 unloaded from Frame 0, Page 5 loaded into Frame 0
Page 3 unloaded from Frame 1, Page 2 loaded into Frame 1
Page 6 unloaded from Frame 2, Page 4 loaded into Frame 2
Page 1 unloaded from Frame 3, Page 6 loaded into Frame 3
Page 5 unloaded from Frame 0, Page 1 loaded into Frame 0
9 page faults
Optimal
Page 4 loaded into Frame 0
Page 3 loaded into Frame 1
Page 4 already in Frame 0
Page 6 loaded into Frame 2
Page 1 loaded into Frame 3
Page 6 already in Frame 2
Page 4 already in Frame 0
Page 3 unloaded from Frame 1, Page 5 loaded into Frame 1
Page 5 unloaded from Frame 1, Page 2 loaded into Frame 1
Page 4 already in Frame 0
Page 6 already in Frame 2
Page 1 already in Frame 3
6 page faults

3
LRU
Page 4 loaded into Frame 0
Page 3 loaded into Frame 1
Page 4 already in Frame 0
Page 6 loaded into Frame 2
Page 1 loaded into Frame 3
Page 6 already in Frame 2
Page 4 already in Frame 0
Page 3 unloaded from Frame 1, Page 5 loaded into Frame 1
Page 1 unloaded from Frame 3, Page 2 loaded into Frame 3
Page 4 already in Frame 0
Page 6 already in Frame 2
Page 5 unloaded from Frame 1, Page 1 loaded into Frame 1
7 page faults