Page replacement algos

After conducting simulations on three page replacement algorithms (Optimal, NRU, Clock), comparing their performance by graphing page faults and disk writes against the number of frames in physical memory yields the following results.

The choice of the refresh rate (N) in the NRU algorithm is important. To determine the optimal N value, We need to balance page faults and disk writes. Therefore, for NRU resets, a refresh rate of fifty memory seemed acceptable based on my two graphs to the left as it gives the most balanced results.

All of these graphs are based on the ls.trace. Reiterating that 50 is my chosen refresh rate for NRU testing based on the two left graphs. The two graphs on the right show page faults vs # of frames and Disk writes vs # of frames for each algorithm. Of Course to no surprise OPT wins in every category. It is our baseline and we are using it to compare NRU and CLOCK.

You can see that in terms of page faults CLOCK consistently beats out NRU coming closer to the impossible OPT algorithm. When analyzing the disk writes it is a bit closer. There are certain times NRU comes out on top as disk writes are important in this algorithm, however not constantly or significantly enough for me to say it is the better algorithm.

Therefore I believe Clock is clearly the winner here if we are baselining off OPT. It beats out NRU and comes close enough to OPT but is actually implementable in a real OS system.