cs4730_assignment3 Jacob Reiser

I implemented a copy of the list with a clockNode structure to keep track of whether or not a page was dirty or not. It is basically a copy of the list.c and list.h files with only that change for use with the clock method.

I implemented the FIFO, LRU and Clock methods for the replacement methods of FIFO, LRU, and Clock. FIFO moves through the list of pages, returning them in sequence of use in a que style. THe LRU simply returns the head of the list for modification, since the head is the most recently added, in a kind of stack. The clock works by checking if a page is 'dirty' or not, and then replacing accordingly, adding a page to the end of the list.

In the pagefault handler, I checked whether or not a page was free before either reading/writing to a free page or finding a new page using the specified replacement type.

The VPN and offset were calculated in MMU by right-shifting the virtual address, then subtracting the right-shifted VPN from the virtual address respectively.

The functionality was tested by generating tests using the input_gen, as well as the provided example inputs, and comparing the results using the vm_reference to the results from my implementations.